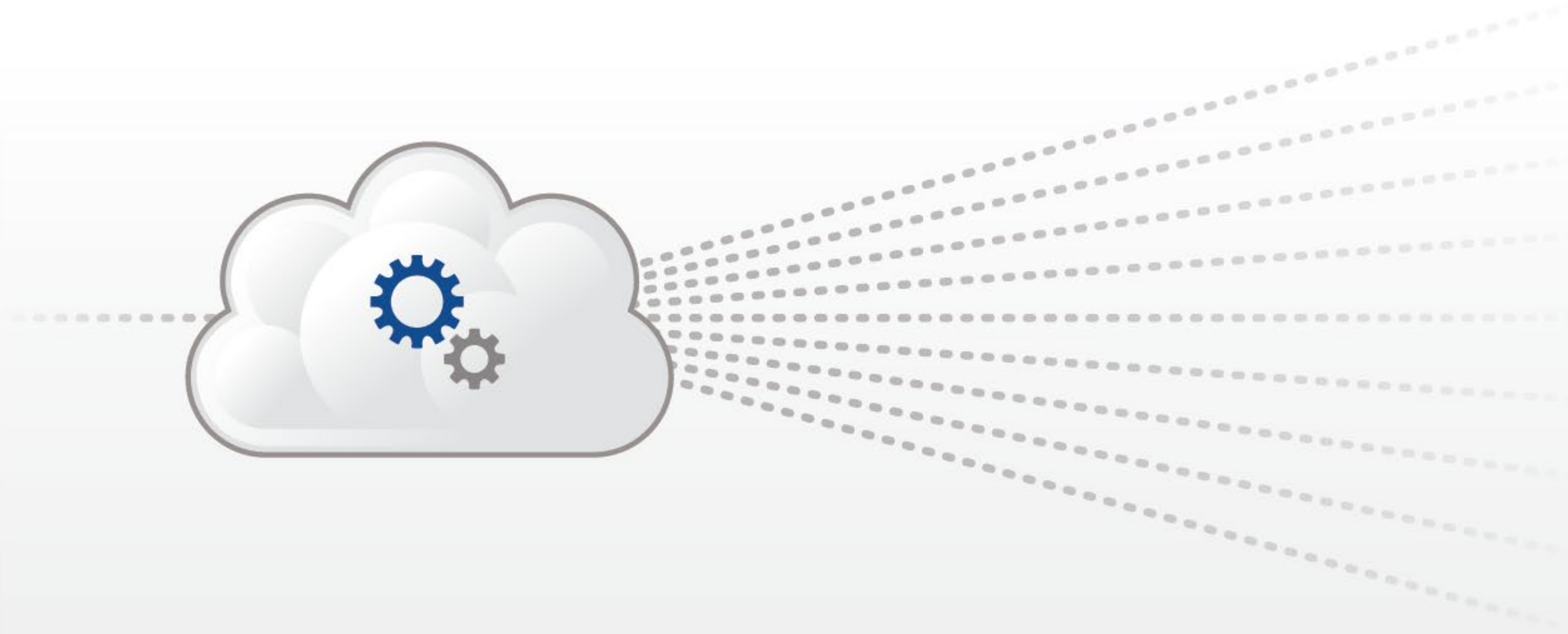


Xalt | Mobility

User Guide

V3 – April 13, 2023

Chapter 8: Gadgets 264

Chapter 9: User Profile Administration 273

Chapter 10: Security 302

Chapter 11: Business Intelligence (BI) 320

Chapter 12: Custom Settings 395

Chapter 13: Image/Asset Management 399

Chapter 14: Dashboards 416

Chapter 15: Import/Export 437

Chapter 16: Tenant Email Server 486

Chapter 1: Concepts

1

What is Xalt | Mobility?

Xalt | Mobility ("Xalt") is a cloud middleware platform that provides secure, real-time access to enterprise data, applications and content on any mobile device. Your firewall must be configured to allow Xalt servers to communicate to your internal systems. Once that link is established, you may use Xalt's App Builder Dashboard to select the data you want to show to one or more end users and build an "app" to access that data. The resulting data objects can be deployed via tablets, smartphones, browsers and desktop widgets. All this can be accomplished without any programming.

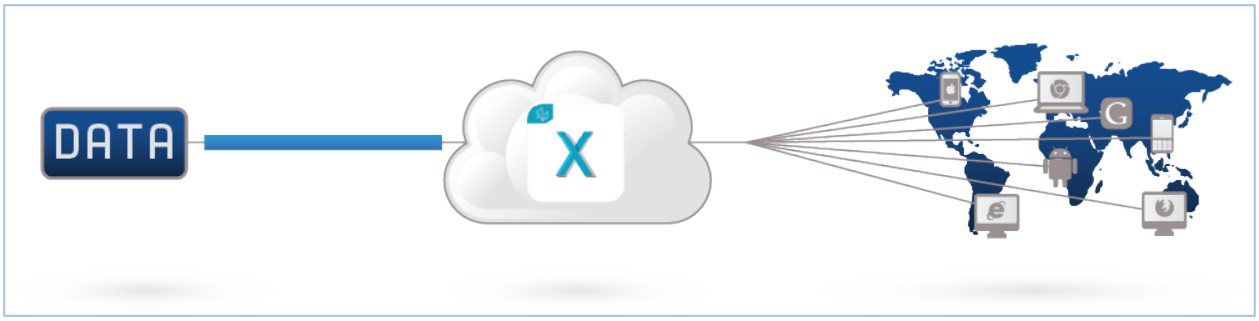


Figure 1: Xalt | Mobility provides Internet-based user interfaces over your enterprise data.

Xalt allows two-way data access, i.e., users can read from and write to your enterprise database. You may create separate data objects for internal and external users, and users with higher or lower security clearance. For example, you may choose to display orders to customers (external users) with prices, quantities and dates. You may show orders to sales reps (internal users) that include sales incentives and bonuses. You may show orders to sales management (higher security users) with costs and profit margins. Similarly, you may allow customers to update the request date and contact information, while sales management can override a price.

Xalt apps are accessed via a thin, light native UI agent that resides on the mobile device, tailored for that device's specific platform, whether iOS, Android, etc. The database remains on your internal system. In addition, Xalt can call your application API's to perform updates. This means that edits, calculated fields, multi-file updates and other application functions are all performed by your internal system.

Connecting Xalt | Mobility to your system(s)

Connecting to your internal system is a two-step process: First, you must download and install the connector gateway. Second, you must define one or more data sources.

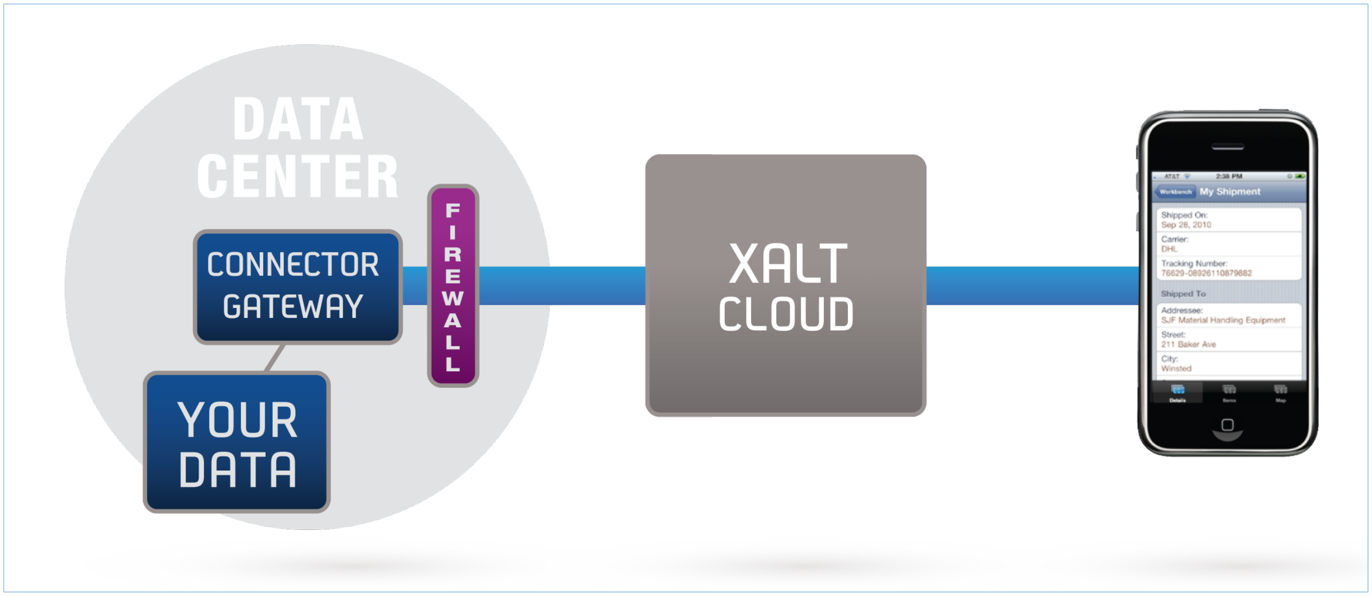
The connector gateway

To protect your internal systems, Hexagon will provide you with a software server to install in your data center. We call this a connector gateway. You will install it on your hardware, which can be either physical or virtual. Locating this server inside your data center provides the following benefits:

- High performance – Xalt can communicate with your internal systems inside the same network, thus reducing chatter between your data center and our cloud servers.

- High security – The connector gateway can be installed inside your firewall or in the DMZ, if you use one. All communications between the connector gateway and the Xalt cloud servers is encrypted. Your internal systems are never open to the Internet at large.

You will need to configure your firewall to allow the Xalt cloud server(s) to communicate with the connector gateway.



Shipped On: Sep 28, 2010
Carrier: DHL
Tracking Number: 76629-08926110879882
Addressee: SJF Material Handling Equipment
Street: 211 Baker Ave
City: Winsted

Figure 2: Connector gateways and data sources

Data Sources

The connection from Xalt to your system is called a data source. Most companies will have multiple data sources between Xalt servers and their internal systems for one or more of the following reasons:

- Connecting to test vs. production environments
- Connecting to different databases (SQL Server, DB2, Oracle…)
- Connecting to systems in different data centers (such as at multiple internal divisions).

There is no limit to the number of data sources you may have.

Data Objects

There are three basic types of user interfaces in Xalt | Mobility: Internet browsers, desktop widgets and mobile devices. Internet browsers and mobile devices provide a menu (called a workbench) of objects to the user. For example, the workbench may include orders, invoices and shipments. Each of these is a data object.

Typically, when a user opens a data object they see a list. To continue our example, they might see a list of orders, invoices or shipments. Each of these may include another object, e.g., the line items. Using order as an example, the order (header) would be a data object, and the order line items would be a second data object. The user would open the order header first and see a list of orders. When the user selects a particular order they would see the line items associated with that order. You can link any number of data objects together, allowing the user to easily navigate from one to the next.

Queries

When the user opens the data object, they are presented with a list. Each list is defined by three components:

- The data shown for each record in the list. For example, a list of orders might show the order number, date, total price, etc.
- The selection criteria or filter that determines whether all the records in the file or table are shown. You may choose to show only open orders, not to mention only those appropriate for the user.
- The sort order of the records. The list of orders may be displayed by when they were entered, newest first.

The collection of these three components is called a query. Each data object must have at least one query, so Xalt will know what to show the user when they open it.

Details

Often the user wants more information about the data object. For example, when looking at a particular order the user may want to see the shipping address. This additional data is called the data object details, and you may provide as much additional data as you like. You may even put it into logical groupings, or sections. For example, the delivery address may be a section. Another section may provide pricing and terms. Another section may provide the list of items on the order. There is no limit to the number of sections, and no limit to the amount of data in each section. There is a practical limit, of course, as too much information will frustrate the users. The best rule is to talk to the users to find out what they want to know, and organize the data accordingly.

Details are optional, and may be unnecessary for data objects with only a small amount of data.

Actions

Query and details are used to define what the user sees when they open a data object. An action defines what they can do with it once it is open. Actions are optional; displaying the data may be all the user needs or is authorized to.

You may think of actions in terms of a menu that is provided to the user. For example, you may allow a customer to update the contact information on an order. You would do this via a menu option that shows them the current contact information and allows them to type over it.

An action can update the database directly or call a program on your internal system. Use actions to call your application API's when specific functions such as edits, field calculations and multi-file updates are involved.

Multiple Data Objects

It is very common to define multiple data objects for a single business object. For example, you may give your customers a data object that shows order information. The query will include a filter that shows only the customer's own orders. You may define a second data object for sales reps that shows orders for all customers in the sales rep's territory. You may define a third data object for sales management that shows costs and profit margins for each order.

The data you show and the actions you allow can all be specified by data object. This is why multiple data objects are often necessary. You do not need to define a separate data object for each customer, however. Xalt allows one data object to work for all customers, while showing only the proper data to each.

Workbenches

Earlier we mentioned that you may provide a menu of data objects to the user. This menu is called a workbench. Workbenches are logical groupings of data objects. There is no limit to the number of data objects on a workbench, and no limit to the number of workbenches you may define.

User Profiles

Each user will be provided with a user profile. User profiles contain what you would expect: the name, ID and password for the user. The user profile is also where you define which workbenches a user is allowed to see.

Some users, usually employees, may have credentials on the internal system. Other users, such as customers or suppliers, may not. You may define both types of users to CX. You can set up user profiles so that the internal system can perform authentication for users it recognizes. For all others, you will allow Xalt to do the authentication.

You can also define user properties such as customer number or warehouse. These properties will be used in the subsetting logic to ensure that each user sees only their own data.

Business Intelligence (BI)

Xalt | Mobility has added the ability to create light Business Intelligence (BI) applications. Hexagon BI allows you to create data sets by connecting to existing SSAS data cubes. You can then create BI Dashboards that connect to these data sets to produce graphs, tables, and details from the dimensions and measurements in the data sets. Xalt | Mobility also integrates directly with Microsoft Power BI to show embedded reports.

Security

Xalt | Mobility employs a Role-based security scheme. All operations in Xalt are authorized to be performed by one or more Security Roles. Likewise, User Profiles can be members of one or more Security Roles. When a user attempts an operation, Xalt security checks the user's effective permissions to see if the user is authorized to perform the operation. Hexagon uses an inclusive method of determining a user's effective permissions. If the user is a member of at least one Security Group that has authorization to the operation, then the user is authorized.

Chapter 2: Connector Gateways & Data Sources

Chapter Summary

The combination of Connector Gateways and Data Sources is used to connect the Xalt Cloud to your data center. A Connector Gateway is a computer in the DMZ network of your data center that runs the Hexagon Connector Gateway software. A data source is a specific connection to a database or enterprise environment. All communications between Hexagon and the back-end system are routed through the data source in the cloud to the Connector Gateway and then ultimately to the back-end system. A single Connector Gateway may host multiple data sources. A data source is associated with one Connector Gateway.

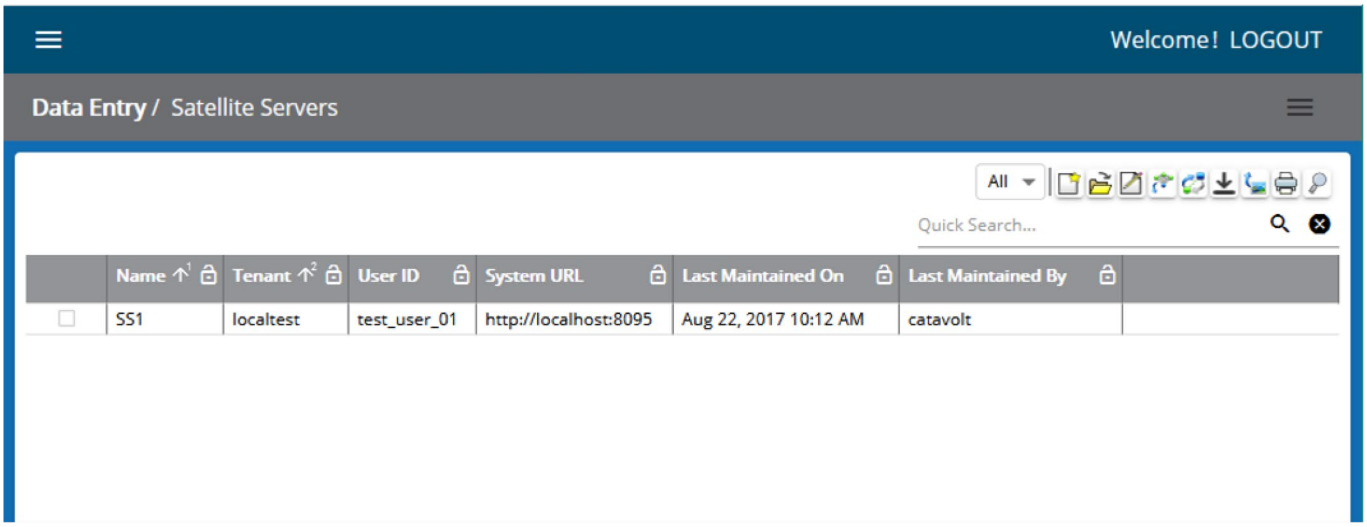
There are settings on data sources that control connection pooling, user credentials and other aspects of the connection. You may create multiple data sources for a single database or ERP environment to tailor the connection settings for certain applications and end users.

Installing a Connector Gateway

To install a Connector Gateway, you must first setup a computer with a supported operating system, hardware and network requirements. Please refer to the Connector Gateway Installation Guide for more information about provisioning and installing a Connector Gateway.

Connector Gateway Administration

You can access the list of Connector Gateways (also referred to as Satellite Servers) that are connected to the Xalt Cloud by signing into Xalt as an application administrator and navigating to the Data Entry Workbench. From there you can create, update and delete Connector Gateways. In addition, there are toolbar and menu actions for testing the connection to the Connector Gateway and refreshing the server's definitions.

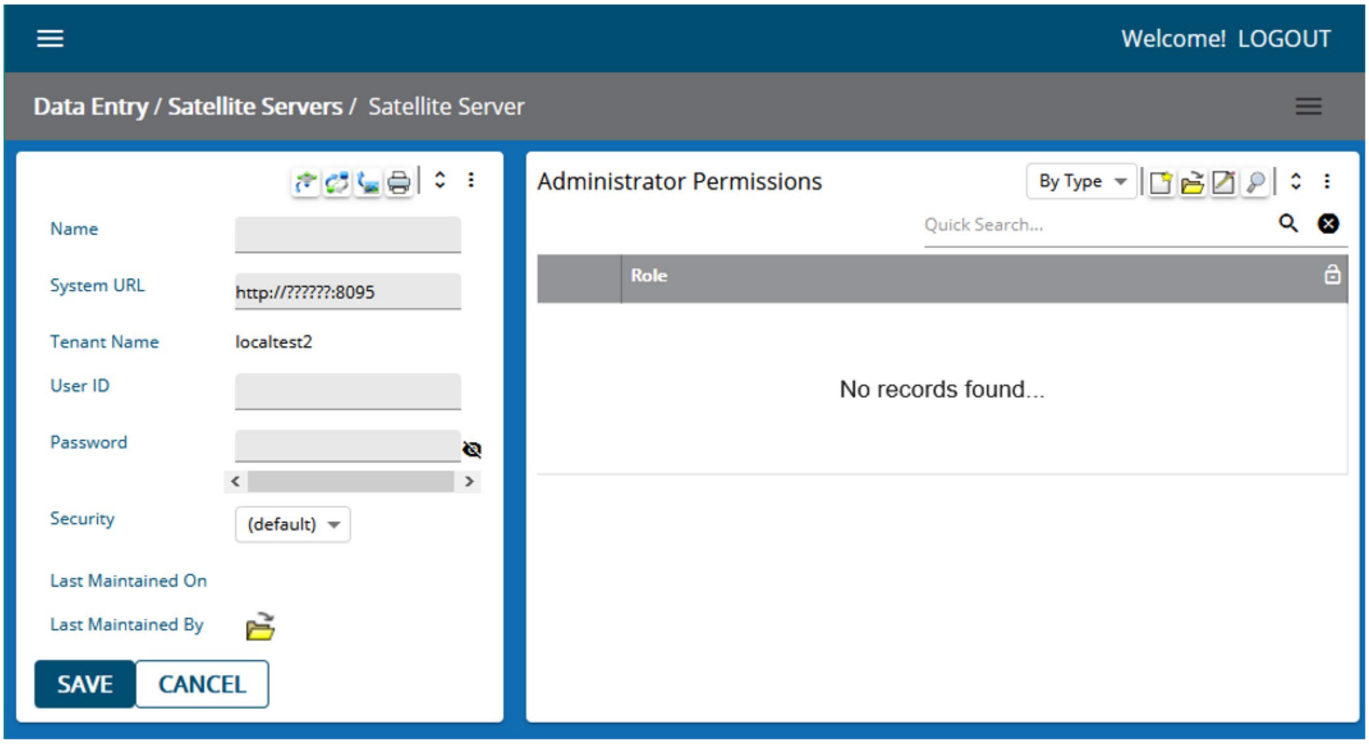


	Name	Tenant	User ID	System URL	Last Maintained On	Last Maintained By	
□	SS1	localtest	test_user_01	http://localhost:8095	Aug 22, 2017 10:12 AM	catavolt	

Figure 1: Connector Gateways list inside of Xalt

Creating Connector Gateways

After installing the Connector Gateway software on the Connector Gateway computer, you must create a Connector Gateway record in Xalt.



Name:
System URL: http://??????:8095
Tenant Name: localtest2
User ID:
Password:
Security: (default)
Last Maintained On
Last Maintained By

Administrator Permissions

No records found...

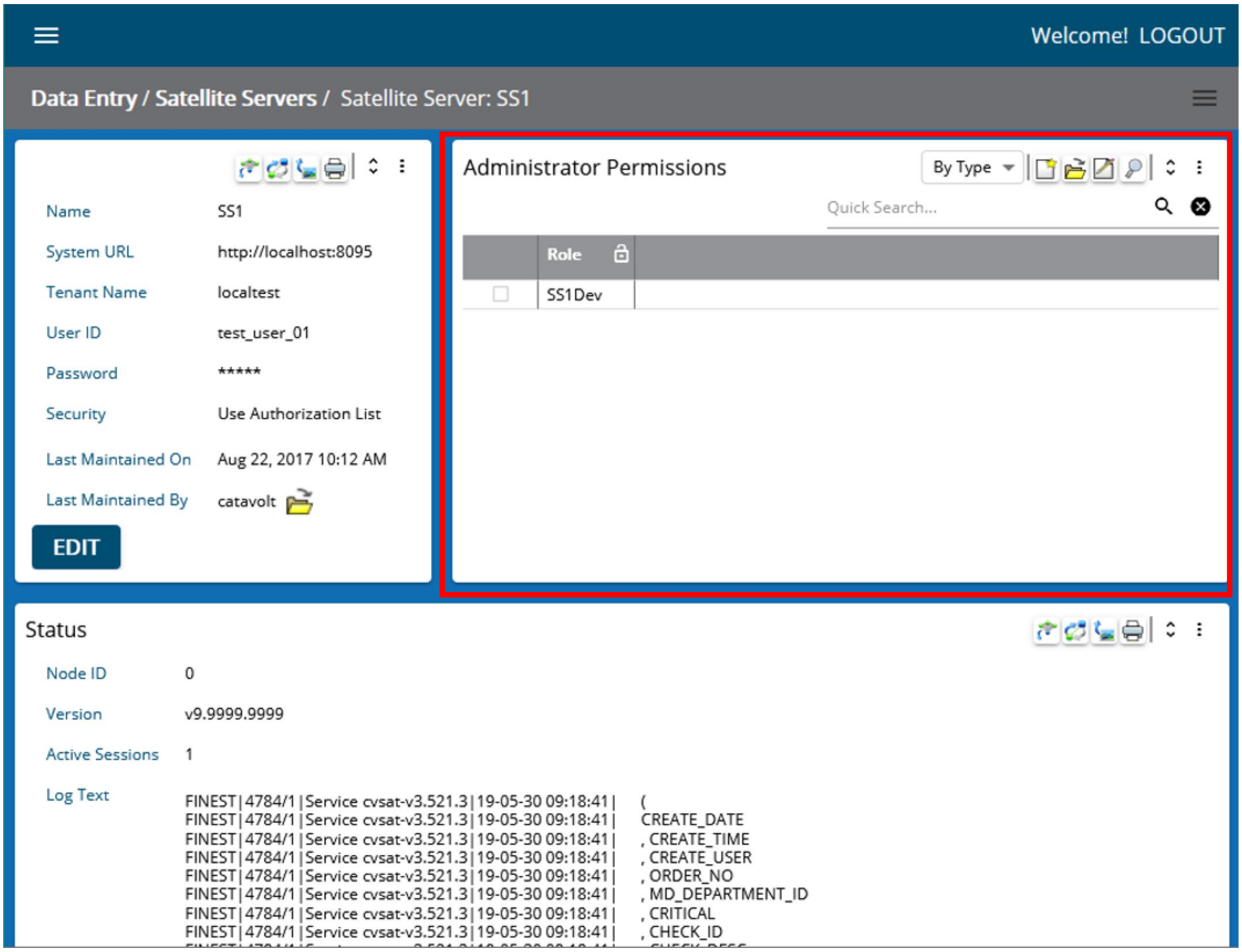
Figure 2: Connector Gateway create screen

When creating the Connector Gateway enter the following properties:

- **Name:** This is the name you will see in the list of Connector Gateways.
- **System URL:** This is the URL that the Xalt cloud will use to communicate with the Connector Gateway. This should use the external IP address of that server.
- **User ID:** Use the user id entered into the configuration file when installing the Connector Gateway software.
- **Password:** Use the password entered into the configuration file when installing the Connector Gateway software.
- **Security:** Specifies who is authorized to create and deploy Xalt applications associated with this Connector Gateway. It is most often used in cases where you have multiple Connector Gateways, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Connector Gateway. The available options are:
    - **(default)** – Use default security for this Connector Gateway. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Connector Gateway.
    - **Use Authorization List** – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Connector Gateway. You can add new Security Roles in the Administrator Permissions section.

Connector Gateway Administrator Permissions

Administrator Permissions is the area where you define the list of Security Roles that can create and deploy Xalt applications on this Connector Gateway. You can add and remove Security Roles to this section if Security is set to Use Authorization List.


Welcome! LOGOUT
Data Entry / Satellite Servers / Satellite Server: SS1
EDIT
Administrator Permissions
By Type
Quick Search...
Status

Name SS1
System URL http://localhost:8095
Tenant Name localtest
User ID test_user_01
Password *****
Security Use Authorization List
Last Maintained On Aug 22, 2017 10:12 AM
Last Maintained By catavolt

	Role
□	SS1Dev

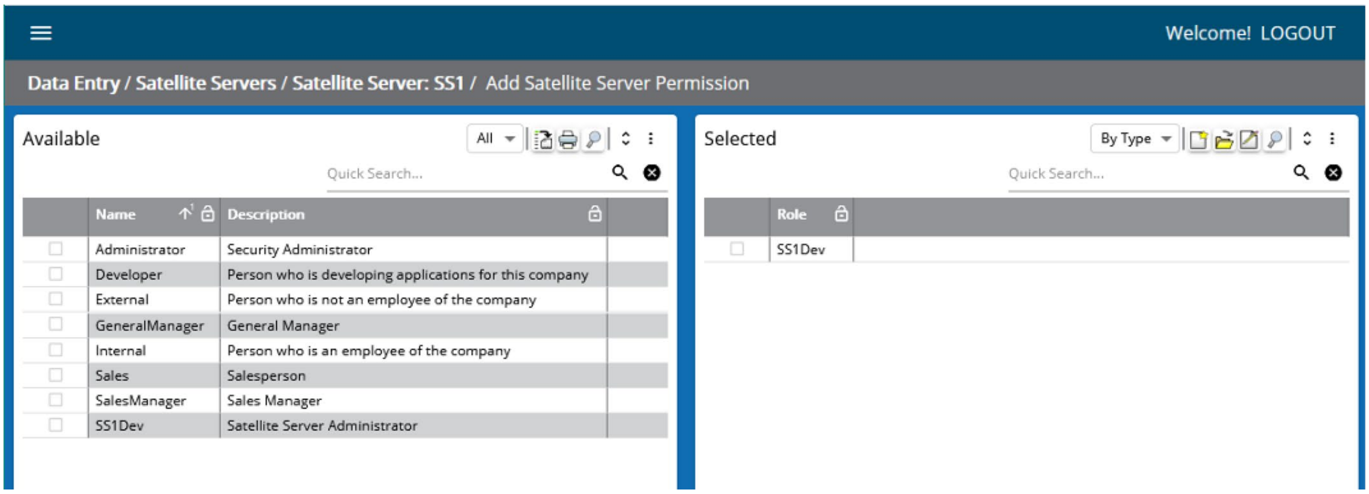
Node ID 0
Version v9.9999.9999
Active Sessions 1
Log Text

```
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   (
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   CREATE_DATE
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CREATE_TIME
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CREATE_USER
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , ORDER_NO
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , MD_DEPARTMENT_ID
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CRITICAL
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CHECK_ID
```

Figure 3: Connector Gateway details with Administrator Permissions section highlighted

When first changing Security from (default) to Use Authorization List, Xalt will default the Administrator Permissions to authorize the Administrator and Developer roles. You are free to remove these existing roles and add new ones.

When adding Administrator Permissions, you will be presented with two lists. The Available Permissions list shows all Security Roles defined in the system. The Selected Permissions list shows the Security Roles that can administer objects associated with this Connector Gateway.



Data Entry / Satellite Servers / Satellite Server: SS1 / Add Satellite Server Permission

Available

	Name	Description
□	Administrator	Security Administrator
□	Developer	Person who is developing applications for this company
□	External	Person who is not an employee of the company
□	GeneralManager	General Manager
□	Internal	Person who is an employee of the company
□	Sales	Salesperson
□	SalesManager	Sales Manager
□	SS1Dev	Satellite Server Administrator

Selected

	Role
□	SS1Dev

Figure 4: The add Connector Gateway Permission view.

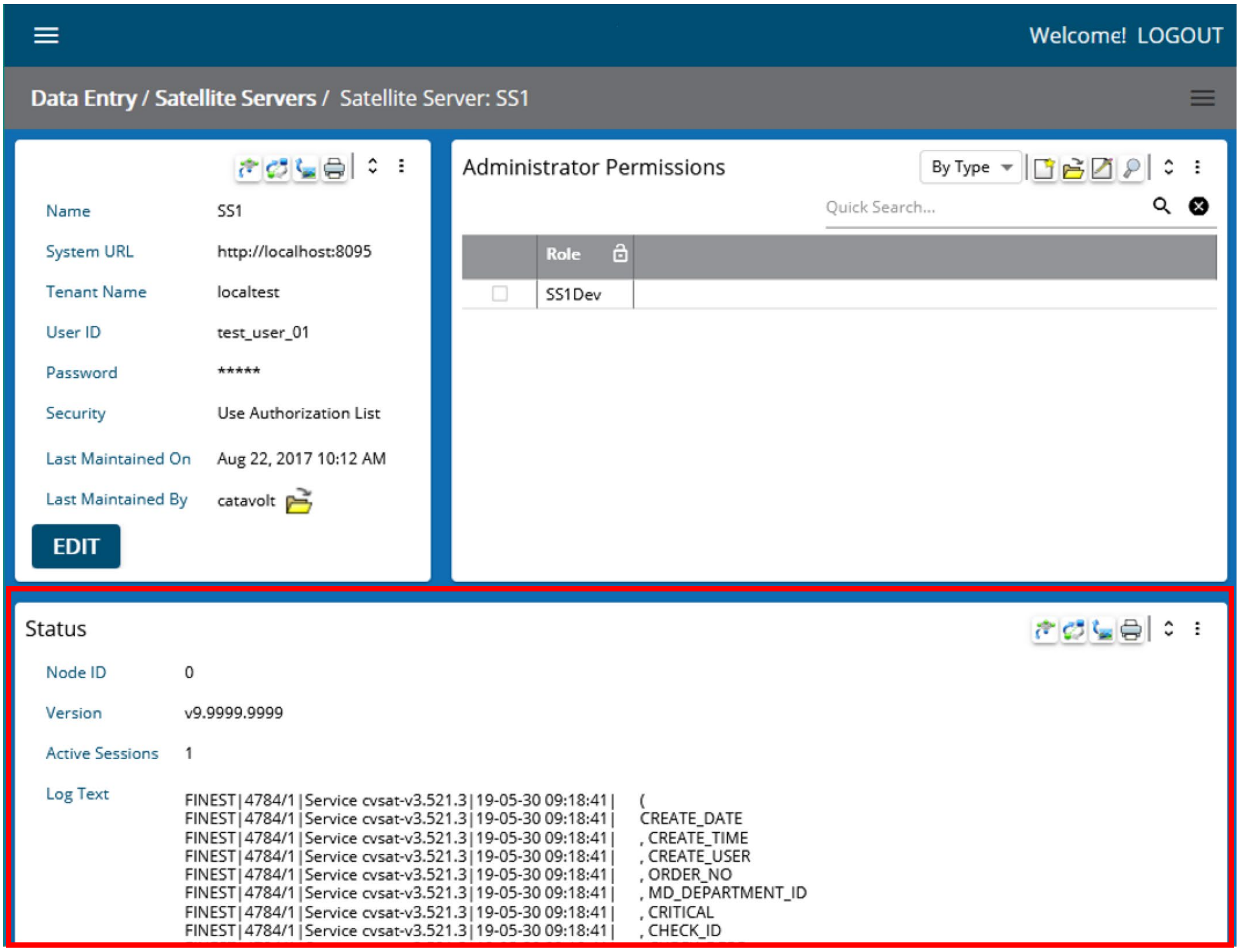
You can select a single or multiple Security Roles and press the Add button to authorize them to the Connector Gateway.

After the Administrator Permission has been created, opening the Administrator Permission record will display the details for the Security Role. See Chapter 10: Security for more information about Security Roles.

Note that you are required to have at least 1 Security Role selected for the Connector Gateway at all times to keep you from getting into a situation where everyone is locked out of administering the Connector Gateway.

Connector Gateway Status

Status is the area that provides read-only current information on this Connector Gateway.



Name SS1
System URL http://localhost:8095
Tenant Name localtest
User ID test_user_01
Password *****
Security Use Authorization List
Last Maintained On Aug 22, 2017 10:12 AM
Last Maintained By catavolt

Administrator Permissions

	Role
□	SS1Dev

Status

Node ID 0
Version v9.9999.9999
Active Sessions 1
Log Text

```
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   (
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   CREATE_DATE
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CREATE_TIME
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CREATE_USER
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , ORDER_NO
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , MD_DEPARTMENT_ID
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CRITICAL
FINEST|4784/1|Service cvsat-v3.521.3|19-05-30 09:18:41|   , CHECK_ID
```

Figure 5: Connector Gateway details with Status section highlighted

- **Node ID:** Specifies the specific load-balanced instance. This will be 0 unless you have multiple load-balanced instances of a Connector Gateway running.
- **Version:** Specifies the version of Connector Gateway software that is currently running.
- **Active Sessions:** Specifies the number of active connections to the Connector Gateway.
- **Log Text:** Shows the last 500 lines of the current Connector Gateway log. If you need to view previous log entries, you can run the **Package Log Files** menu action on the Connector Gateway list (see below for more details).

Test Connector Gateway Connection

You can test the connection between the Xalt cloud and the Connector Gateway at any time by using the Test Connector Gateway Connection action from the toolbar or context menu. Selecting a Connector Gateway and invoking this action will test the connection between the Xalt Cloud and the Connector Gateway.

Package Log Files

The Connector Gateway maintains multiple log files that can be used by Hexagon Support to diagnose issues. The Package Log Files menu option will show a list of available log files on the Connector Gateway. You can select one or more files from the list and choose the Download Log Files menu option to zip them up and download them to your local computer.

Connector Gateway Cache

Information is sent to the Connector Gateway as you define data sources and data objects that are associated with the server. The Connector Gateway uses this information to access the back-end system. This information can be lost when the Connector Gateway software is reinstalled, the Connector Gateway is installed on a different computer or by any other event that causes the folders that contain this information to be deleted. You can use the Update Connector Gateway Information action on the toolbar and context menu to rewrite this information to the Connector Gateway. You will be prompted to re-enter Data Source password information, as these values only reside on the Connector Gateway.

Data Sources

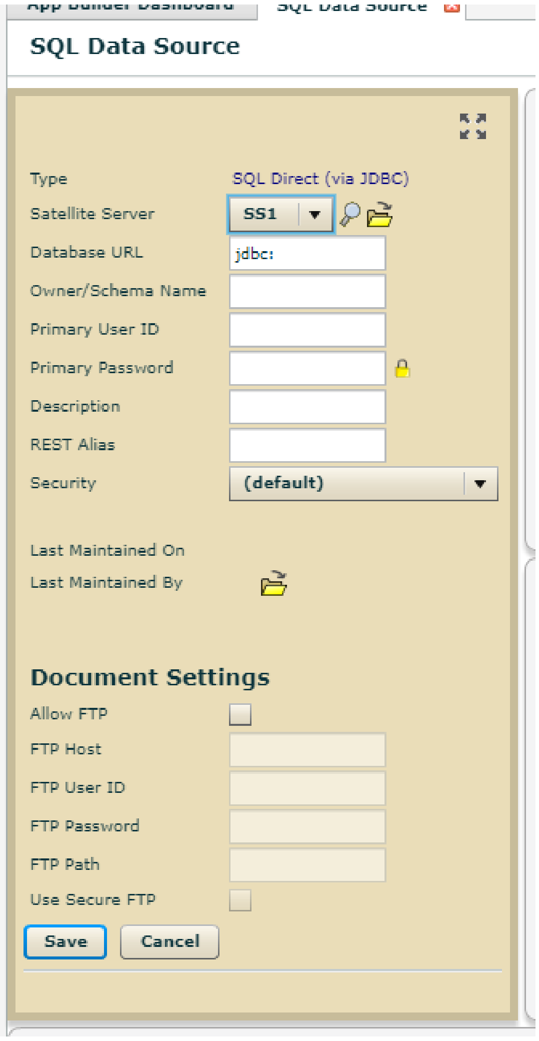
A Data Source is a pointer to a single database or ERP environment. Each data source is associated with a connector gateway. When creating a data source, you will be prompted to enter the information about the database or ERP environment you would like to connect to in addition to the credentials that will be used to connect.

There are several types of data sources. Some data sources connect directly to a database while others connect to an API that is specific to a back-end system. The properties that you must specify when maintaining a data source are dependent on the type of data source you have selected.

Data Source Type: SQL Direct

The SQL Direct data source is used to connect directly to any database that supports Java JDBC connectivity, which includes Microsoft SQL Server, IBM DB2, Oracle, PostgreSQL, MySQL and many others. Please check with the database vendor for more information about their JDBC support. When connecting to a database directly Xalt will create SQL dynamically based on your application definition to access data.

When creating a SQL Direct data source, you must specify the following properties. All properties are required.



SQL Data Source

Type	SQL Direct (via JDBC)
Satellite Server	SS1
Database URL	jdbc:
Owner/Schema Name	
Primary User ID	
Primary Password	
Description	
REST Alias	
Security	(default)
Last Maintained On	
Last Maintained By	

Document Settings

Allow FTP	□
FTP Host	
FTP User ID	
FTP Password	
FTP Path	
Use Secure FTP	□

Figure 6: SQL Direct Data Source create screen

Type

You select this when you click the 'New' menu option.

Connector gateway

This is the connector gateway that will be used to communicate with the database.

Database URL

This is the JDBC URL that will be used by the connector gateway to connect to the database. The URL will contain the host name or IP address of the database, the type of database, the database name and various settings. The host name or IP address must be one that is accessible by the connector gateway on the connector gateway's network. The database does not have to be accessible directly by the Xalt cloud. JDBC URL syntax is dependent on the database vendor. Some examples of database URLs can be found below. In each example the type of database is in purple, the IP address or host name is in green, the database name is in blue and the properties required by Xalt are in red.

- IBM DB2 : jdbc:as400://serverip;naming=sql;extended dynamic=true;translate binary=true;errors=full;prompt=false;date format=iso
- Microsoft SQL Server : jdbc:sqlserver://serverip;databaseName=dbname
- Postgresql : jdbc:postgresql://serverip/dbname

Owner/ Schema

For most databases this is the collection in which the tables are stored. For Microsoft SQL Server this value is usually defaulted to dbo. For PostgreSQL this value is usually the same as the user id. For DB2 this is the library in which the tables are located. Please refer to the database vendor to find out more about table owners and database schemas for your database.

Primary User ID

This is the user id which will be used to access the database. Other user IDs can be used for certain objects as described in the Connection Profiles section. The primary user id is the one that is used when no alternate Connection Profile is specified and during the application building and design activities.

Primary Password

This is the password that will be used along with the primary user ID. Description: This is the description that will be used inside of Xalt to identify this data source. Note that this value is only stored on the Connector gateway and cannot be retrieved once entered.

Description

The description of this data source that will be used to identify it in Xalt.

REST Alias

The REST Alias for this Data Source. This is used when accessing the Xalt | Mobility REST API. More information about this option can be found in the **Xalt | Mobility REST API Guide**.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- (default) – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.
- Use Authorization List – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

Document Settings

These settings control how external files can be downloaded using an FTP server

Allow FTP

This setting controls whether attachments are supported for this data source.

FTP Host

The network name or IP address of the FTP host that will be used by the connector gateway to establish an FTP session. This name or IP address has to be accessible by the connector gateway and does not have to be directly accessible by the Xalt cloud. If your FTP server is using an alternate port, you may add a colon and the port number to this value.

FTP User ID/ Password

The user id and password that will be used to connect to the FTP host.

FTP Path

This is the path that will be used as the root folder for all documents attached using this data source. The path should be in FTP syntax with / as folder separators.

Use Secure FTP

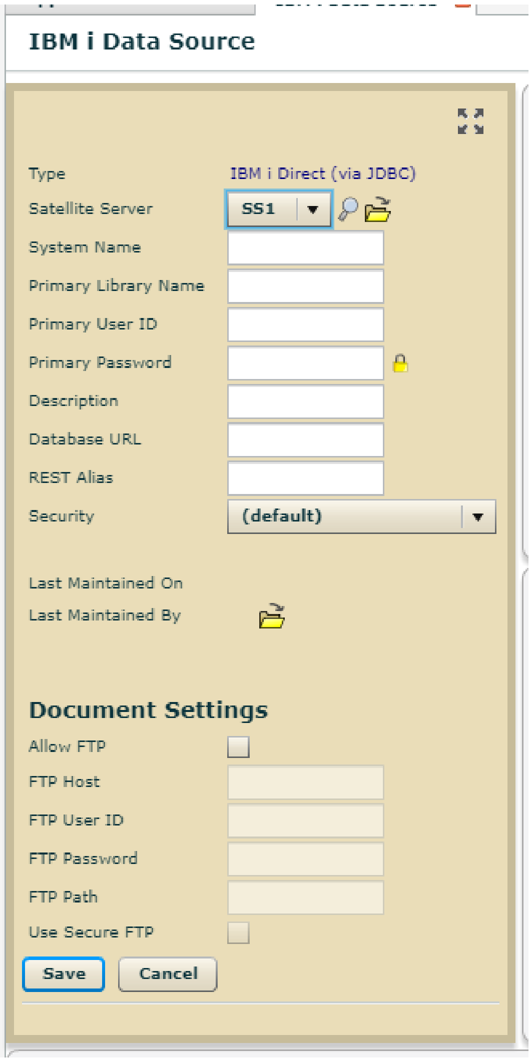
Check this setting to use FTPS connections. FTPS uses SSL to encrypt all communications between the connector gateway and the FTP host.

Data Source Type: IBMi Direct

A special type of data source is available for accessing IBM DB2 databases that reside on a IBM i server. The IBM i Direct data source uses JDBC and shares many characteristics with the SQL Direct data source. The IBM i data source supports the IBM i notion of a library list and the ability to call IBM i RPG and CL programs.

When creating an IBM i data source, you must enter a primary library name. This library will be used to control the library list for the data source. Upon establishing a connection to the database the IBM i data source will look for a data area named INLIBL in the primary library to read the library list for the connection. This data area must exist in the primary library, must be accessible by the primary user ID and must contain a list of library names separated by blanks. There are no positional requirements for the library names, meaning that you do not have to pad the names with blanks to 10 characters. The list of libraries in INLIBL will be used as your library list, so include all libraries that you may need (e.g. QTEMP, QGPL, your primary library, etc).

When creating an IBM i data source you must provide the following properties:



IBM i Data Source

Type	IBM i Direct (via JDBC)
Satellite Server	SS1
System Name	
Primary Library Name	
Primary User ID	
Primary Password	
Description	
Database URL	
REST Alias	
Security	(default)
Last Maintained On	
Last Maintained By	

Document Settings

Allow FTP	□
FTP Host	
FTP User ID	
FTP Password	
FTP Path	
Use Secure FTP	□

Figure 7: IBM i Direct Data Source create screen

Type

You select this when you click the 'New' menu option

Connector gateway

This is the connector gateway that will be used to communicate with the database

System Name

This is the system name or IP address of the IBM i server. This name or IP address has to be accessible from the selected connector gateway and does not have to be accessible from the Xalt cloud.

Primary Library Name

This is the IBM i library that contains the INLIBL data area

Primary User ID/ Primary Password

The user ID and password used to connect to the IBM i server.

Description

The description of this data source that will be used to identify it in Xalt.

Database URL

This is the JDBC URL that will be used to connect to the database. You can leave this blank and have the system default it for you based on the system name.

REST Alias

The REST Alias for this Data Source. This is used when accessing the Xalt | Mobility REST API. More information about this option can be found in the **Xalt | Mobility REST API Guide**.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- (default) – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.
- Use Authorization List – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

Document Settings

These settings control how external files can be downloaded using an FTP server

Allow FTP

This setting controls whether attachments are supported for this data source.

FTP Host

The network name or IP address of the FTP host that will be used by the connector gateway to establish an FTP session. This name or IP address has to be accessible by the connector gateway and does not have to be directly accessible by the Xalt cloud. If your FTP server is using an alternate port, you may add a colon and the port number to this value.

FTP User ID/ Password

The user id and password that will be used to connect to the FTP host.

FTP Path

This is the path that will be used as the root folder for all documents attached using this data source. The path should be in FTP syntax with / as folder separators.

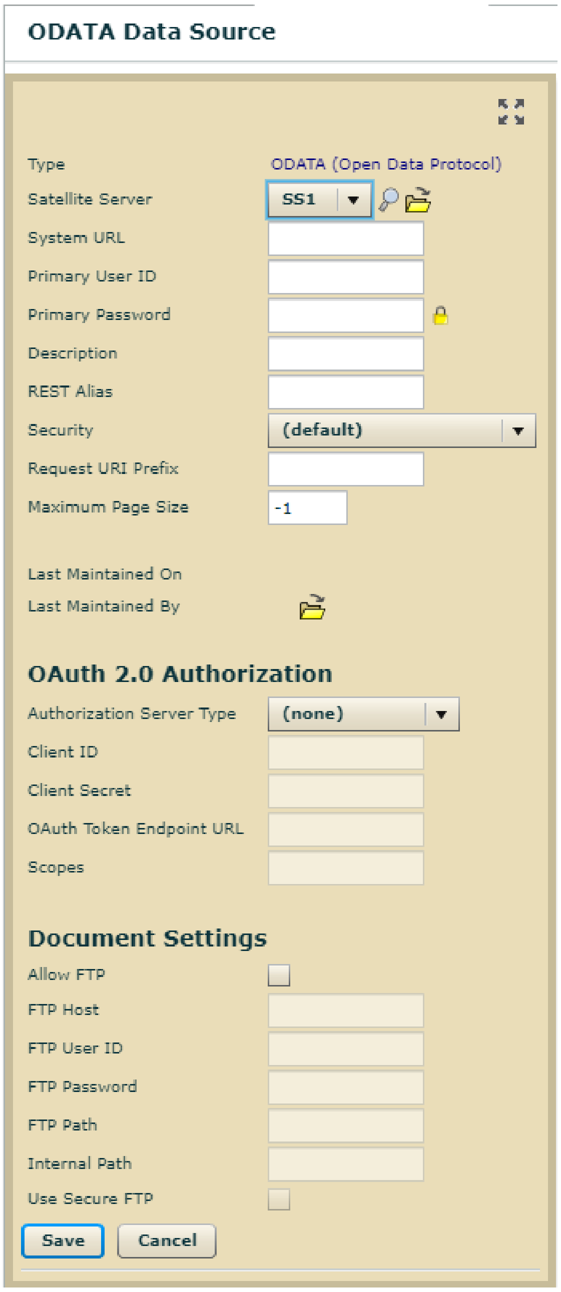
Use Secure FTP

Check this setting to use FTPS connections. FTPS uses SSL to encrypt all communications between the connector gateway and the FTP host.

Data Source Type: ODATA (Open Data Protocol)

The ODATA data source has been built to use the ODATA $metadata API to access data.

When creating an ODATA data source, you must provide the following properties.



ODATA Data Source

Type	ODATA (Open Data Protocol)
Satellite Server	SS1
System URL	
Primary User ID	
Primary Password	
Description	
REST Alias	
Security	(default)
Request URI Prefix	
Maximum Page Size	-1
Last Maintained On	
Last Maintained By	

OAuth 2.0 Authorization

Authorization Server Type	(none)
Client ID	
Client Secret	
OAuth Token Endpoint URL	
Scopes	

Document Settings

Allow FTP	□
FTP Host	
FTP User ID	
FTP Password	
FTP Path	
Internal Path	
Use Secure FTP	□

Figure 8: ODATA Data Source create screen

Type

You select this when you click the 'New' menu option

Connector gateway

This is the connector gateway that will be used to communicate with the SAP service

System URL

This is the base URL for the published ODATA spec. Xalt will add "/$metadata" to the end of this URL in order to "discover" the available metadata for the web service implementation.

Primary User ID/ Primary Password

Used in conjunction with OAuth 2.0 Authentication section (see below).

Description

The description of this data source that will be used to identify it in Xalt.

REST Alias

The REST Alias for this Data Source. This is used when accessing the Xalt | Mobility REST API. More information about this option can be found in the **Xalt | Mobility REST API Guide**.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- (default) – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.
- Use Authorization List – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

Request URI Prefix

This feature was added to support the Hexagon PPM multi-tenant ODATA protocol and is not used with regular ODATA Data Sources

Maximum Page Size

[Note: Starting at Satellite Server version 3.591.0, this value is no longer used and will be disabled. It will be removed from the product all together at a future time].

Xalt clients will request enough records to fill one or more screens (e.g 75 records). In cases where the ODATA backend server has a small maximum record size request, this may result in only a subset of records for a list being returned. If your ODATA server has a small maximum record request size, you can enter the value here. Xalt will then attempt to make multiple back-end requests to fulfill a single client record request. Set the value to -1 to have Xalt perform default processing.

Authorization Server Type

The type of Authorization Server used to authenticate Web Service calls. The allowed values are:

- (none) – Do not use an Authorization server. This option will use Basic Authentication or no authentication (see Authorization section below).
- Hexagon PPM – Specifies an SPF Authorization server (see Authorization section below).
- Other – Specifies a generic OAuth Authorization server (see Authorization section below).

Client ID

The Client ID as registered in the Authorization Server (note the Client ID should be configured to use the flow that Xalt will use (see Authorization section below).

Client Secret

The Client Secret as defined in the Authorization Server.

OAuth Token Endpoint URL

The URL used to access the Token Endpoint. This is required if Authorization Server Type is "Other".

Scopes

The list of scopes to use when requesting an access token. This is only available if Authorization Server Type is "Other".

Document Settings

These settings control how external files can be downloaded using an FTP server

Allow FTP

This setting controls whether attachments are supported for this data source.

FTP Host

The network name or IP address of the FTP host that will be used by the connector gateway to establish an FTP session. This name or IP address has to be accessible by the connector gateway and does not have to be directly accessible by the Xalt cloud. If your FTP server is using an alternate port, you may add a colon and the port number to this value.

FTP User ID/ Password

The user id and password that will be used to connect to the FTP host.

FTP Path

This is the path that will be used as the root folder for all documents attached using this data source. The path should be in FTP syntax with / as folder separators.

Use Secure FTP

Check this setting to use FTPS connections. FTPS uses SSL to encrypt all communications between the connector gateway and the FTP host.

Authorization

Xalt will use determine the type of authorization to use as follows:

1) Authorization Server Type = (none). In this case, Xalt looks at the Primary User ID/Primary Password. If these values are specified, Xalt will use Basic Authorization in the Web Service request header. If these values are not specified, Xalt will omit the Authorization header (e.g. for public web services)
2) Authorization Server Type = Hexagon PPM. In this case, Xalt will use the System URL specified to "discover" the OAuth Server information per SmartAPI rules. Xalt will add "/description" in order to find the following information:

"TokenIssuerUri": "http://ppm.intergraph.com/spfnpapcfgsvc/spfauthentication/oauth"
"TokenAudience": "EE9C5479-A52E-4D11-80AE-BFDDCE9A603F" <===== Resource
"SupportedScopes": ["ingr.api"]

Xalt will then add "/.well-known/openid-configuration" to the TokenIssuerUri in order to find the following value:

"token_endpoint":http://ppm.intergraph.com/oauth/connect/token

3) Authorization Server Type = Other. In this case, the user must specify the OAuth Token Endpoint URL (required) and Scopes (optional) (same as "token_endpoint" and "SupportedScopes" values discovered in Hexagon PPM)

In both case 2) and 3), the user is required to specify the Client ID and Client Secret. If they also specify a Primary User ID/Primary Password, Xalt will use the **password** grant_type:

grant_type=password&username=UUUU&password=PPPP&client_id=IIIII&client_secret=SSSS&scope=XXXXX&resource=RRRRRR (resource only specified if found in option 2).

If they do not specify a Primary User ID/Primary Password, Xalt will use the **client_credentials** grant_type (Client ID / Client Secret are passed as Basic Authorization in request header):

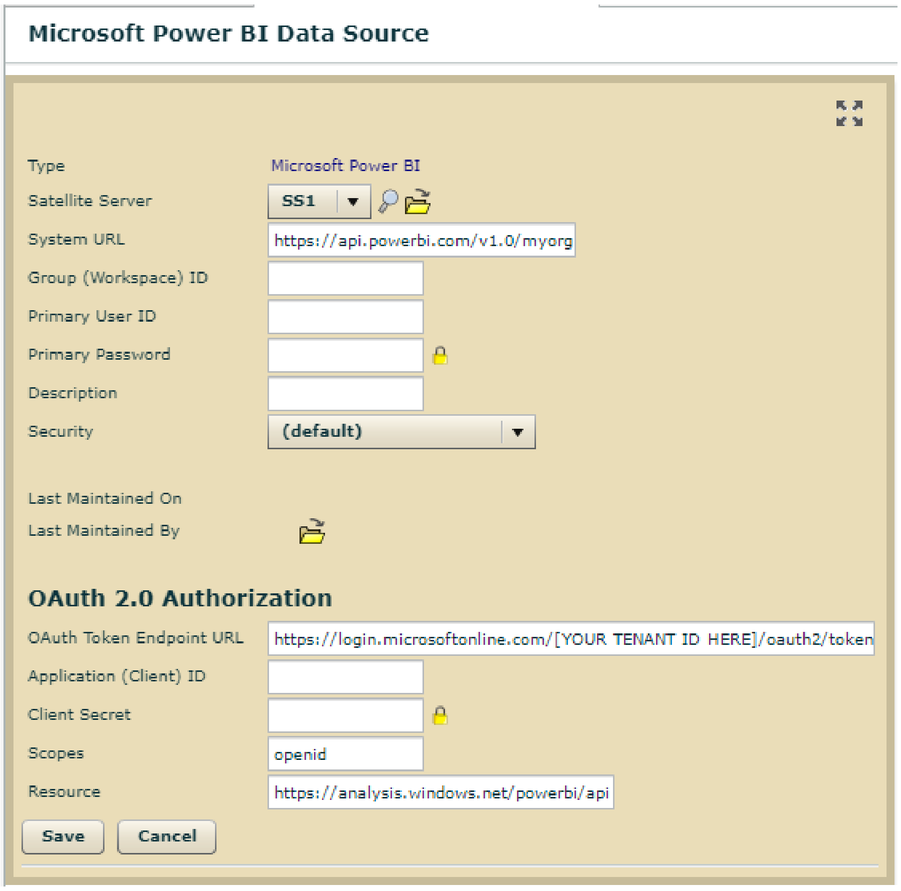
grant_type=client_credentials&scope=XXXXX&resource=RRRRRR (resource only specified if found in option 2).

Note that the Access token granted above is used when querying meta data. If the Data Object being queried/updated has a Connection Profile of (data source), the Access Token will also be used for these requests as well. If the Data Object being queried/updated has a Connection Profile of (current user), the Access Token of the logged in user will be used instead.

Data Source Type: Microsoft Power BI

This data source has been built to access Microsoft Power BI reports embedded via Azure Active Directory.

When creating a Power BI data source, you must provide the following properties.



Microsoft Power BI Data Source

Type	Microsoft Power BI
Satellite Server	SS1
System URL	https://api.powerbi.com/v1.0/myorg
Group (Workspace) ID	
Primary User ID	
Primary Password	
Description	
Security	(default)
Last Maintained On	
Last Maintained By	

OAuth 2.0 Authorization

OAuth Token Endpoint URL	https://login.microsoftonline.com/[YOUR TENANT ID HERE]/oauth2/token
Application (Client) ID	
Client Secret	
Scopes	openid
Resource	https://analysis.windows.net/powerbi/api

Figure 9: Microsoft Power BI Data Source create screen

Type

You select this when you click the 'New' menu option

Connector gateway

This is the connector gateway that will be used to communicate with the Power BI service

System URL

This is the base URL for the Power BI Service. Xalt will default this value to https://api.powerbi.com/v1.0/myorg. It will not need to be changed under normal circumstances.

Primary User ID/Primary Password

Used in conjunction with OAuth 2.0 Authentication section (see below).

Description

The description of this data source that will be used to identify it in Xalt.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- (default) – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.

- Use Authorization List – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

OAuth Token Endpoint URL

The URL used to access the Token Endpoint. Xalt defaults this value to . https://login.microsoftonline.com/[YOUR TENANT ID HERE]/oauth2/token. You need to replace [YOUR TENANT ID HERE] with your Azure AD Tenant ID.

Application (Client) ID

The Client ID as defined in your Azure AD installation.

Client Secret

The Client Secret as defined in your Azure AD installation.

Scopes

The list of scopes to use when requesting an access token. Xalt defaults this value to "openid". It will not need to be changed under normal circumstances.

Resource

The resource to use when requesting an access token. Xalt defaults this value to "https://analysis.windows.net/powerbi/api". It will not need to be changed under normal circumstances.

Authorization

Xalt will use the following to determine the type of authorization to use:

The user is required to specify Client ID and Client Secret. If they also specify a Primary User ID/Primary Password, Xalt will use the **password** grant_type:

grant_type=password&client_id=[app_(client)_id]&client_secret=[client_secret]&username=[primary_user]&password=[primary_password]&resource=https://analysis.windows.net/powerbi/api&scope=openid

This corresponds to the "User-Owns-Data" Power BI embedding model. The primary user specified should own the Workspace you plan to embed reports from. Note this user must not have additional security verification (two-factor authentication) turned on.

If you do not specify a Primary User ID/Primary Password, Xalt will use the **client_credentials** grant_type:

grant_type=client_credentials&client_id=[app_(client)_id]&client_secret=[client_secret]&
resource=https://analysis.windows.net/powerbi/api&scope=openid

This corresponds to the "App-Owns-Data" Power BI embedding model. The service principal for the specified Application ID should have Admin authority to the Workspace you plan to embed reports from.

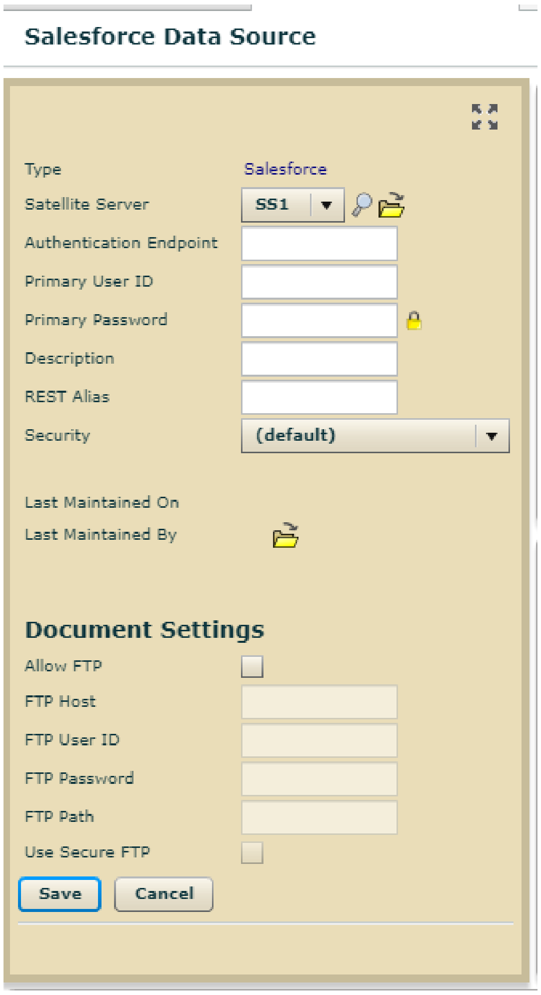
Note that the Access token granted above is used when querying meta data. If the Data Object being queried/updated has a Connection Profile of (data source), the Access Token will also be used for these requests as well. If the Data Object being queried/updated has a Connection Profile of (current user), the Access Token of the logged in user will be used instead.

For more information on configuring a Power BI Data Source, see Appendix F: Microsoft Power BI Integration.

Data Source Type: Salesforce

The Salesforce data source has been built to use Salesforce's SOAP API to access data. The Salesforce data source uses Salesforce's Partner Web Services WSDL via Java proxies.

When creating a Salesforce data source, you must provide the following properties.



Salesforce Data Source

Type	Salesforce
Satellite Server	SS1 ▾
Authentication Endpoint	
Primary User ID	
Primary Password	
Description	
REST Alias	
Security	(default) ▾
Last Maintained On	
Last Maintained By	

Document Settings

Allow FTP	☐
FTP Host	
FTP User ID	
FTP Password	
FTP Path	
Use Secure FTP	☐

Figure 10: Salesforce Data Source create screen

Type

You select this when you click the 'New' menu option

Connector gateway

This is the connector gateway that will be used to communicate with the database

Authentication Endpoint

This is the URL that will be used to authenticate the user/password when connecting to Salesforce. Note that the **partner** authentication endpoint URL should be used. The partner URL has a format similar to: (https://login.salesforce.com/services/Soap/u/25.0/).

Primary User ID/ Primary Password

The user ID and password used to connect to Salesforce.

Description

The description of this data source that will be used to identify it in Xalt.

REST Alias

The REST Alias for this Data Source. This is used when accessing the Xalt | Mobility REST API. More information about this option can be found in the Xalt | Mobility REST API Guide.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- (default) – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.
- Use Authorization List – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

Document Settings

These settings control how external files can be downloaded using an FTP server

Allow FTP

This setting controls whether attachments are supported for this data source.

FTP Host

The network name or IP address of the FTP host that will be used by the connector gateway to establish an FTP session. This name or IP address has to be accessible by the connector gateway and does not have to be directly accessible by the Xalt cloud. If your FTP server is using an alternate port, you may add a colon and the port number to this value.

FTP User ID/ Password

The user id and password that will be used to connect to the FTP host.

FTP Path

This is the path that will be used as the root folder for all documents attached using this data source. The path should be in FTP syntax with / as folder separators.

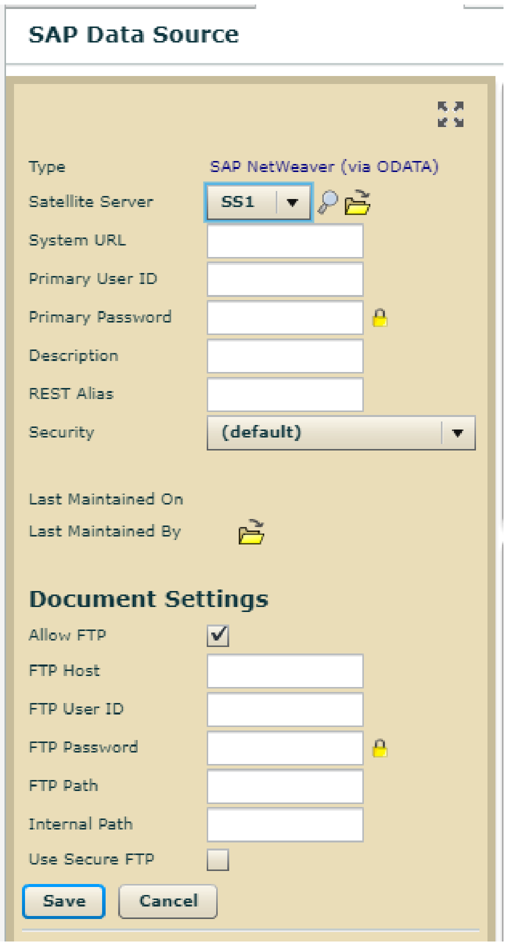
Use Secure FTP

Check this setting to use FTPS connections. FTPS uses SSL to encrypt all communications between the connector gateway and the FTP host.

Data Source Type: SAP NetWeaver (via ODATA)

The SAP data source has been built to use Salesforce's NetWeaver ODATA API to access data.

When creating an SAP data source, you must provide the following properties.



SAP Data Source

Type	SAP NetWeaver (via ODATA)
Satellite Server	SS1
System URL	
Primary User ID	
Primary Password	
Description	
REST Alias	
Security	(default)
Last Maintained On	
Last Maintained By	

Document Settings

Allow FTP	☑
FTP Host	
FTP User ID	
FTP Password	
FTP Path	
Internal Path	
Use Secure FTP	□

Figure 11: SAP Data Source create screen

Type

You select this when you click the 'New' menu option

Connector gateway

This is the connector gateway that will be used to communicate with the SAP service

System URL

This is the URL that will be used to connect to the published SAP service and access data. The System URL has a format similar to: (https://sapes1.sapdevcenter.com/sap/opu/odata/sap/ZCD204_EPM_DEMO_SRV).

Primary User ID/ Primary Password

The user ID and password used to connect to the SAP service.

Description

The description of this data source that will be used to identify it in Xalt.

Document Settings

These settings control how external files can be downloaded using an FTP server

REST Alias

The REST Alias for this Data Source. This is used when accessing the Xalt | Mobility REST API. More information about this option can be found in the **Xalt | Mobility REST API Guide**.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- (default) – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.
- Use Authorization List – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

Allow FTP

This setting controls whether attachments are supported for this data source.

FTP Host

The network name or IP address of the FTP host that will be used by the connector gateway to establish an FTP session. This name or IP address has to be accessible by the connector gateway and does not have to be directly accessible by the Xalt cloud. If your FTP server is using an alternate port, you may add a colon and the port number to this value.

FTP User ID/ Password

The user id and password that will be used to connect to the FTP host.

FTP Path

This is the path that will be used as the root folder for all documents attached using this data source. The path should be in FTP syntax with / as folder separators.

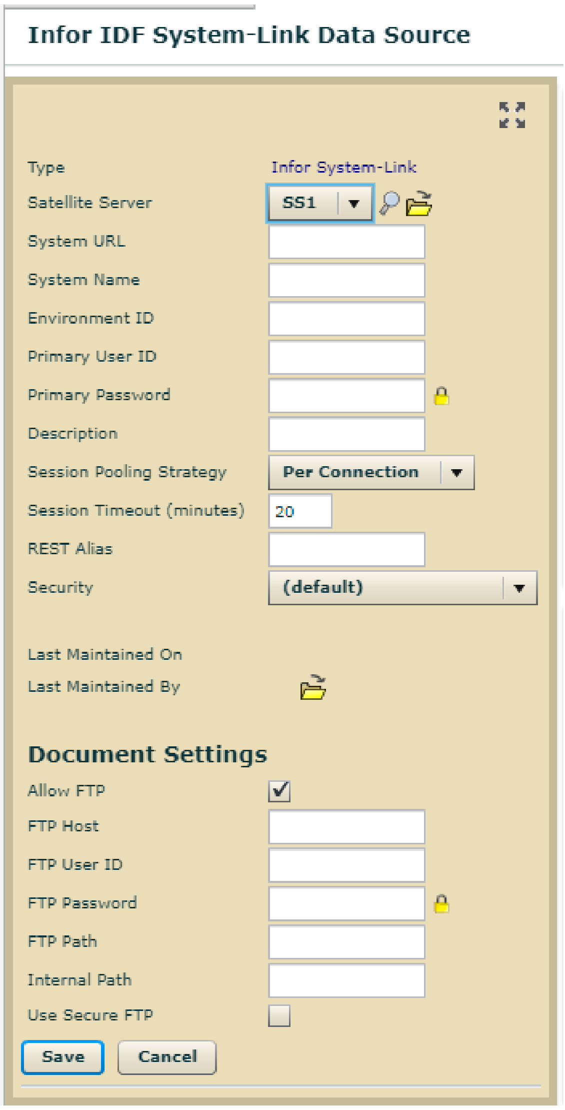
Use Secure FTP

Check this setting to use FTPS connections. FTPS uses SSL to encrypt all communications between the connector gateway and the FTP host.

Data Source Type: Infor IDF System-Link

The Infor IDF System-Link data source has been built to use the Infor IDF System-Link API to access data. No direct connection is made to the database. Instead requests are sent to the IDF system via XML and responses are sent back in XML. The response XML is then parsed and processed by Xalt. The Infor IDF System Link data source uses all of the existing Infor Business Objects and transactions in addition to any user created business objects and transactions created with the Infor Enterprise Integrator product. The Infor IDF System-Link data source also supports IDF style attachments and text objects.

When creating an Infor IDF System-Link data source, you must provide the following properties.



Infor IDF System-Link Data Source

Type	Infor System-Link
Satellite Server	SS1
System URL	
System Name	
Environment ID	
Primary User ID	
Primary Password	
Description	
Session Pooling Strategy	Per Connection
Session Timeout (minutes)	20
REST Alias	
Security	(default)
Last Maintained On	
Last Maintained By	

Document Settings

Allow FTP	☑
FTP Host	
FTP User ID	
FTP Password	
FTP Path	
Internal Path	
Use Secure FTP	☐

Figure 12: Infor IDF System-Link Data Source create screen

Type

You select this when you click the 'New' menu option

Connector gateway

This is the connector gateway that will be used to communicate with the database

System URL

This is the URL that the System-Link request should be sent to. In most cases this value should be http://<servername>:36001/SystemLink/servlet/SystemLinkServlet. The <servername> should be the IP address or network name that is accessible from the selected connector gateway and does not have to be accessible directly by the Xalt cloud.

System Name

This is the system name as known by System-Link. Please refer to an example System-Link request from the Power-Link Navigation menu for the exact text.

Environment ID

The two--character IDF environment id.

Primary User ID/ Primary Password

The user ID and password used to perform a System--Link login to the IDF environment.

Description

The description of this data source that will be used to identify it in Xalt.

Session Pooling Strategy

This option controls how System-Link sessions are reused. The available options are

- **Per Connection:** Each end user session will use a single System-Link session and will attempt to reuse the session for all requests for this data source.
- **No Pooling:** Each System-Link request is made on a new System-Link session. System-Link sessions are not reused.

Session Timeout (minutes)

This setting only applies to Per Connection session pooling strategy and controls the maximum span of time in minutes that a System-Link session can exist while it is idle. Each request made on a System-Link session resets the timeout counter. When a session does timeout, a new session is automatically created for any future requests.

REST Alias

The REST Alias for this Data Source. This is used when accessing the Xalt | Mobility REST API. More information about this option can be found in the **Xalt | Mobility REST API Guide**.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- **(default)** – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.
- **Use Authorization List** – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

Document Settings

These settings control how attachments work with System-Link

Allow FTP

This setting controls whether attachments are supported for this data source.

FTP Host

The network name or IP address of the FTP host that will be used by the connector gateway to establish an FTP session. This name or IP address has to be accessible by the connector gateway and does not have to be directly accessible by the Xalt cloud. If your FTP server is using an alternate port, you may add a colon and the port number to this value.

FTP User ID/ Password

The user id and password that will be used to connect to the FTP host.

FTP Path

This is the path that will be used as the root folder for all documents attached using this data source. The path should be in FTP syntax with / as folder separators.

Internal Path

This is the path that will be used to create attachment and media file records in the IDF environment. This path will be used by Power-Link to open the attachments when users are in Power-Link. In most situations this should be a windows style network path that is accessible from Power-Link.

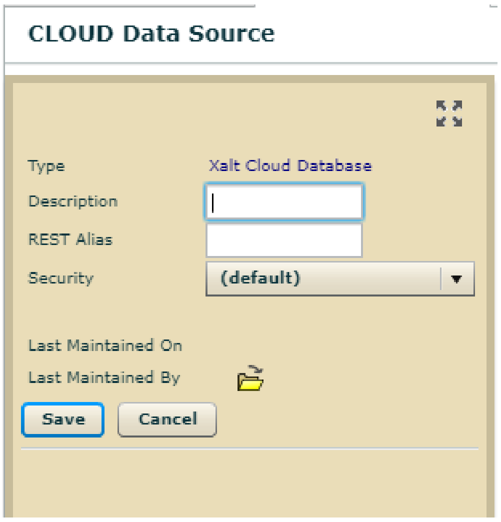
Use Secure FTP

Check this setting to use FTPS connections. FTPS uses SSL to encrypt all communications between the connector gateway and the FTP host.

Data Source Type: Xalt Cloud Database

The Xalt Cloud Database data source is used to access data for the Xalt Metrics application.

When creating an Xalt Cloud Database data source, you must specify the following properties. All properties are required.



CLOUD Data Source

Type: Xalt Cloud Database
Description:
REST Alias:
Security: (default)

Last Maintained On
Last Maintained By

Figure 13: Xalt Cloud Database Data Source create screen

Type

You select this when you click the 'New' menu option.

Description

The description of this data source that will be used to identify it in Xalt.

REST Alias

The REST Alias for this Data Source. This is used when accessing the Xalt | Mobility REST API. More information about this option can be found in the Xalt | Mobility REST API Guide.

Security

Specifies who is authorized to create and deploy Xalt applications associated with this Data Source. It is most often used in cases where you have multiple Data Sources, (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Data Source. The available options are:

- **(default)** – Use default security for this Data Source. Users who are members of the Administrator and/or Developer Security Roles will be able to administer applications on this Data Source.
- **Use Authorization List** – Specify a list of Security Roles that are able to create and deploy Xalt applications on this Data Source. You can add new Security Roles in the Administrator Permissions section.

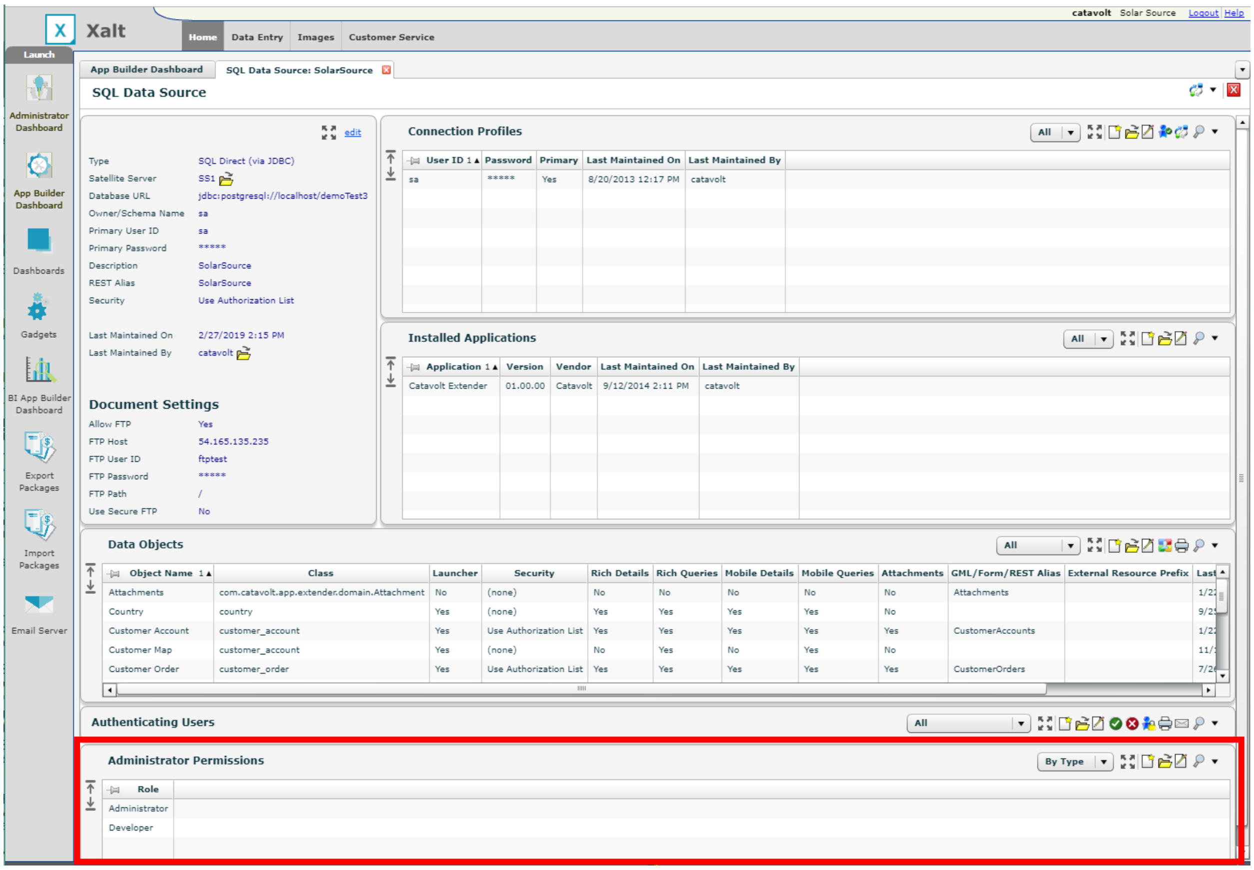
Data Source Connection Profiles

A data source connection profile is a user ID and password that is used by the data source to connect to the back end system. A data source requires at least one connection profile as its primary user id and password. The primary connection profile is used by the data source during application design and administration. During the runtime the primary connection profile is used unless otherwise setup on the Data Object being accessed.

You can create multiple connection profiles. These profiles can then be used when configuring data objects.

Data Source Administrator Permissions

Administrator Permissions is the area where you define the list of Security Roles that can create and deploy Xalt applications for this Data Source. You can add and remove Security Roles to this section if Security is set to Use Authorization List.



SQL Data Source

Type	SQL Direct (via JDBC)
Satellite Server	SS1
Database URL	jdbc:postgresql://localhost/demoTest3
Owner/Schema Name	sa
Primary User ID	sa
Primary Password	*****
Description	SolarSource
REST Alias	SolarSource
Security	Use Authorization List
Last Maintained On	2/27/2019 2:15 PM
Last Maintained By	catavolt

Document Settings

Allow FTP	Yes
FTP Host	54.165.135.235
FTP User ID	ftptest
FTP Password	*****
FTP Path	/
Use Secure FTP	No

Connection Profiles

User ID 1 ▲	Password	Primary	Last Maintained On	Last Maintained By
sa	*****	Yes	8/20/2013 12:17 PM	catavolt

Installed Applications

Application 1 ▲	Version	Vendor	Last Maintained On	Last Maintained By
Catavolt Extender	01.00.00	Catavolt	9/12/2014 2:11 PM	catavolt

Data Objects

Object Name 1 ▲	Class	Launcher	Security	Rich Details	Rich Queries	Mobile Details	Mobile Queries	Attachments	GML/Form/REST Alias	External Resource Prefix	Last
Attachments	com.catavolt.app.extender.domain.Attachment	No	(none)	No	No	No	No	No	Attachments		1/2
Country	country	Yes	(none)	Yes	Yes	Yes	Yes	No			9/2
Customer Account	customer_account	Yes	Use Authorization List	Yes	Yes	Yes	Yes	Yes	CustomerAccounts		1/2
Customer Map	customer_account	Yes	(none)	No	Yes	No	Yes	No			11/
Customer Order	customer_order	Yes	Use Authorization List	Yes	Yes	Yes	Yes	Yes	CustomerOrders		7/2

Authenticating Users

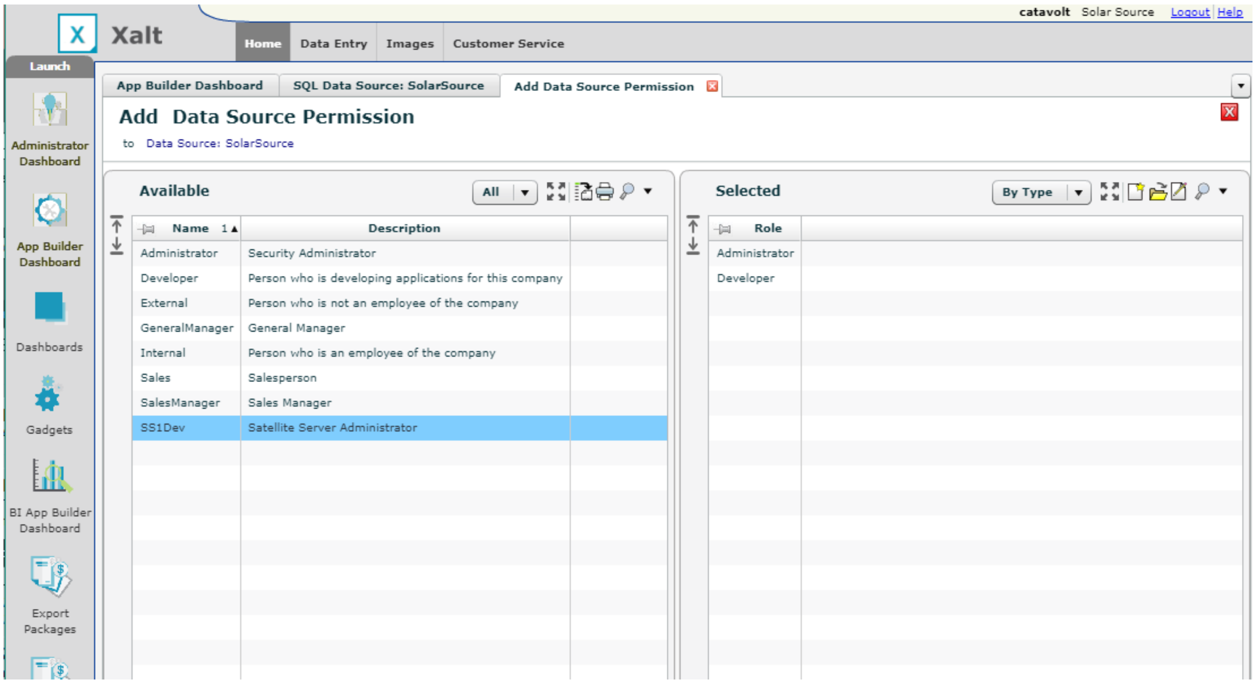
Administrator Permissions

Role
Administrator
Developer

Figure 14: Data Source details with Administrator Permissions section highlighted

When first changing Security from (default) to Use Authorization List, Xalt will default the Administrator Permissions to authorize the Administrator and Developer roles. You are free to remove these existing roles and add new ones.

When adding Administrator Permissions, you will be presented with two lists. The Available Permissions list shows all Security Roles defined in the system. The Selected Permissions list shows the Security Roles that can administer objects associated with this Data Source.



Add Data Source Permission

to Data Source: SolarSource

Name 1 ▲	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	Satellite Server Administrator

Role
Administrator
Developer

Figure 15: The Add Data Source Permission View

You can select a single or multiple Security Roles and press the Add button to authorize them to the Data Source.

After the Administrator Permission has been created, opening the Administrator Permission record will display the details for the Security Role. See Chapter 10: Security for more information about Security Roles.

Note that you are required to have at least 1 Security Role selected for the Data Source at all times to keep you from getting into a situation where everyone is locked out of administering the Data Source.

Authenticating Users

Authenticating Users displays all User Profiles that are using this Data Source as an Authentication Service.

Test Data Source Connection

A data source contains information about connecting to a back-end system. Also, the data objects that are associated with the data source contain metadata for the back end system. Users may experience undesirable behavior if the configuration of the data source and data objects is not consistent with the back end system. You can perform the Test Data Source Connection menu action to verify that the Data Source can successfully connect to the back-end system.

Reset Cached Meta Data

Certain metadata that describes the back end system is cached for performance considerations. An example of this metadata is the list of available columns in a back-end database table. Sometimes this cache can be out of synch with the back end system. This usually occurs when the back end system has changed, for example when new columns are added to a database table. There is an action on the Data Source context menu named Reset Cached

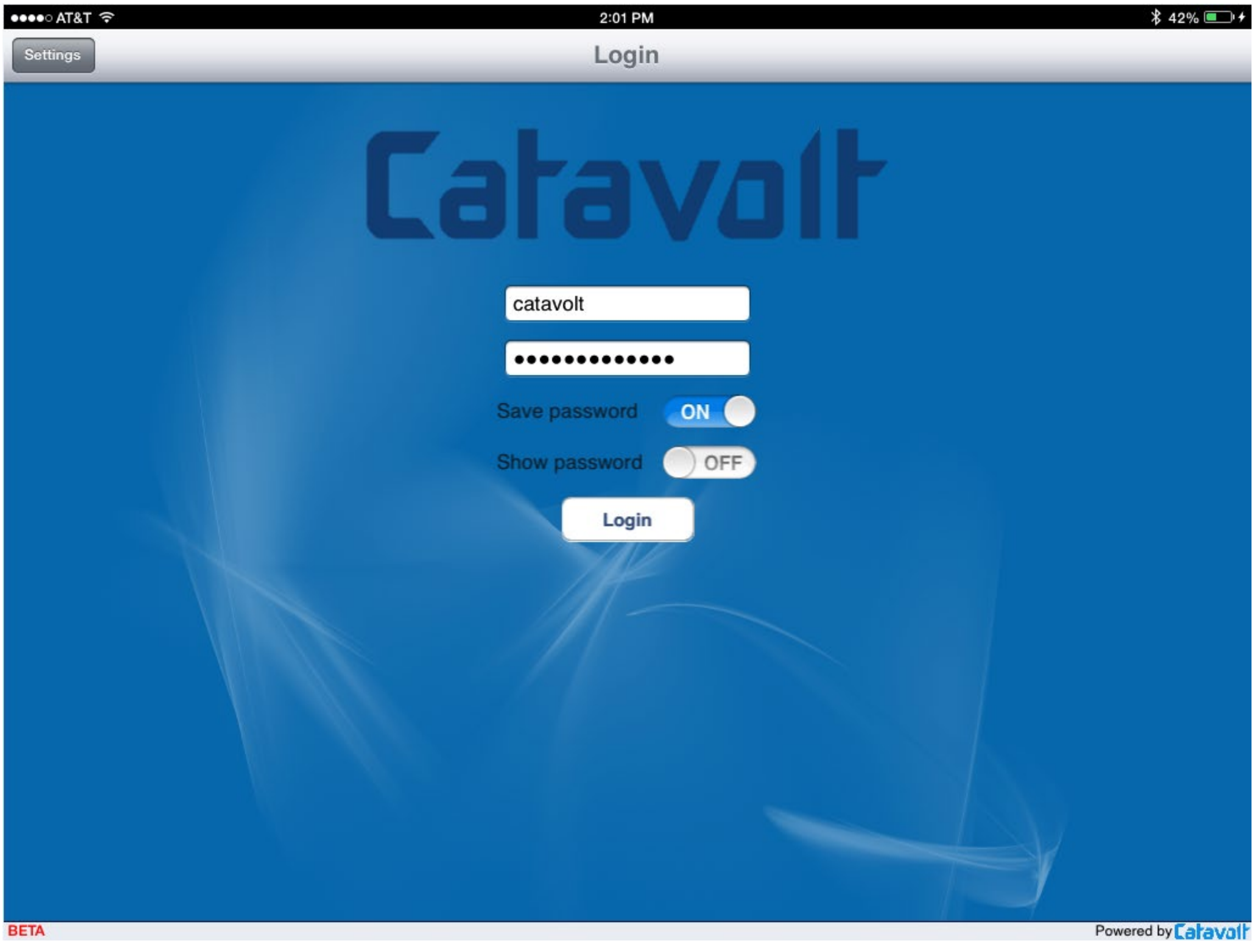
Meta Data that can be used to reset all cached metadata for a data source. The metadata will then be reloaded as needed as you continue to use Xalt.

Upload Image

Images can be used to make the end user experience more intuitive and to create good looking screens. You can use your own images in Xalt by using the Upload Image action on the data source toolbar and context menu. This action will present you with a prompt in which you can select the image file to upload. The image that is uploaded is then available to use across all data sources. Images can be used for launchers and menu actions. Launcher images are displayed in a 48px square area. Menu action images are displayed in a 16px square area. Uploaded images are scaled automatically so that they fit in these areas. It is recommended to use images that closely match the dimensions in which they will be displayed. You can also view, upload, and remove images from a dedicated launcher. See Chapter 13: Image/Asset Management for more information.

Uploading Mobile Branding Materials

You can use your own company logo when accessing Xalt using your mobile device by using the Upload Branding Materials action on the data source context menu. This action will present you with a prompt in which you can select the branding files to upload. Changing branding will affect how your login screen and main workbench window appear on mobile devices:



catavolt

••••••••••••••

Save password ON

Show password OFF

Figure 16: Mobile device Xalt login screen after customer-specific branding has been applied.

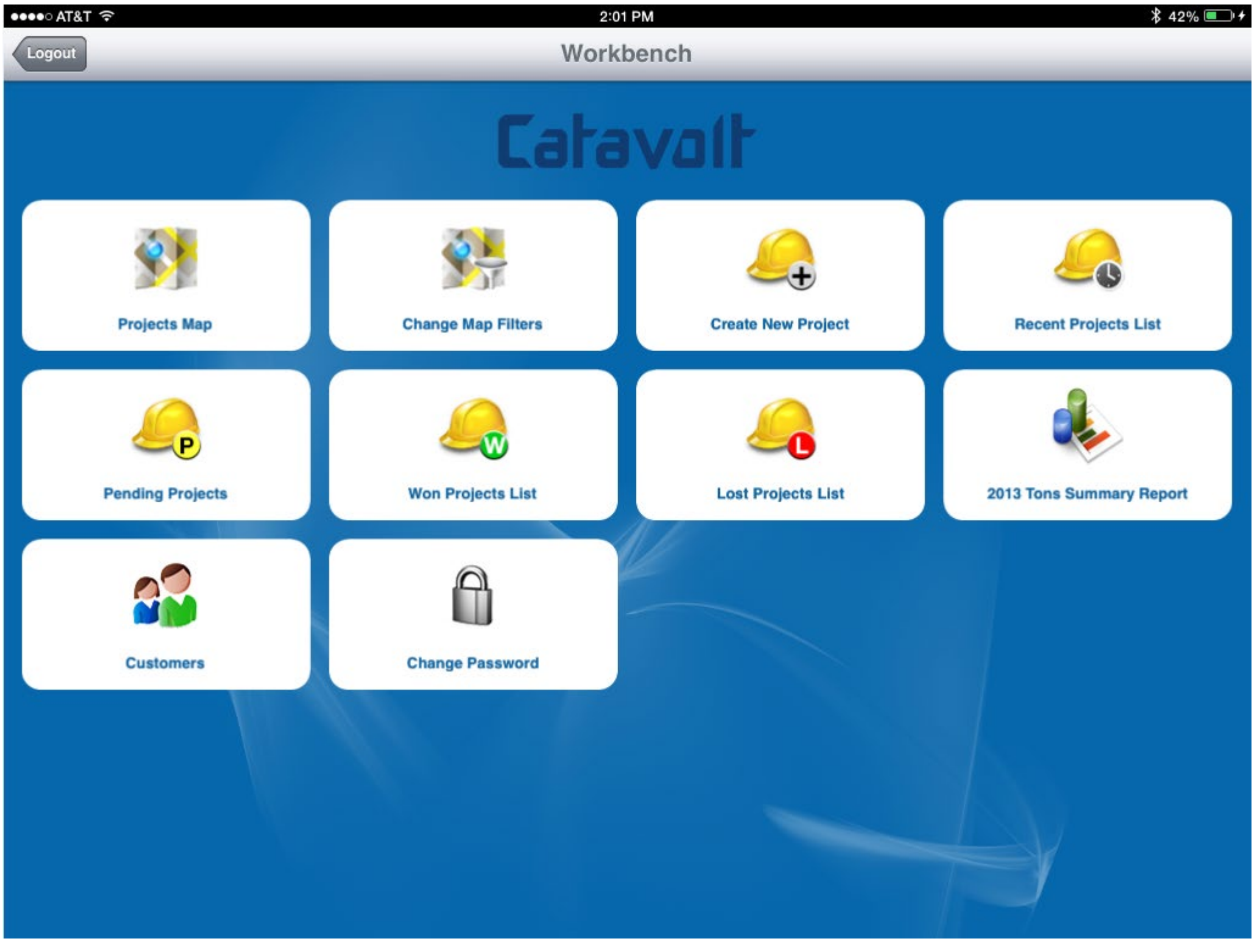


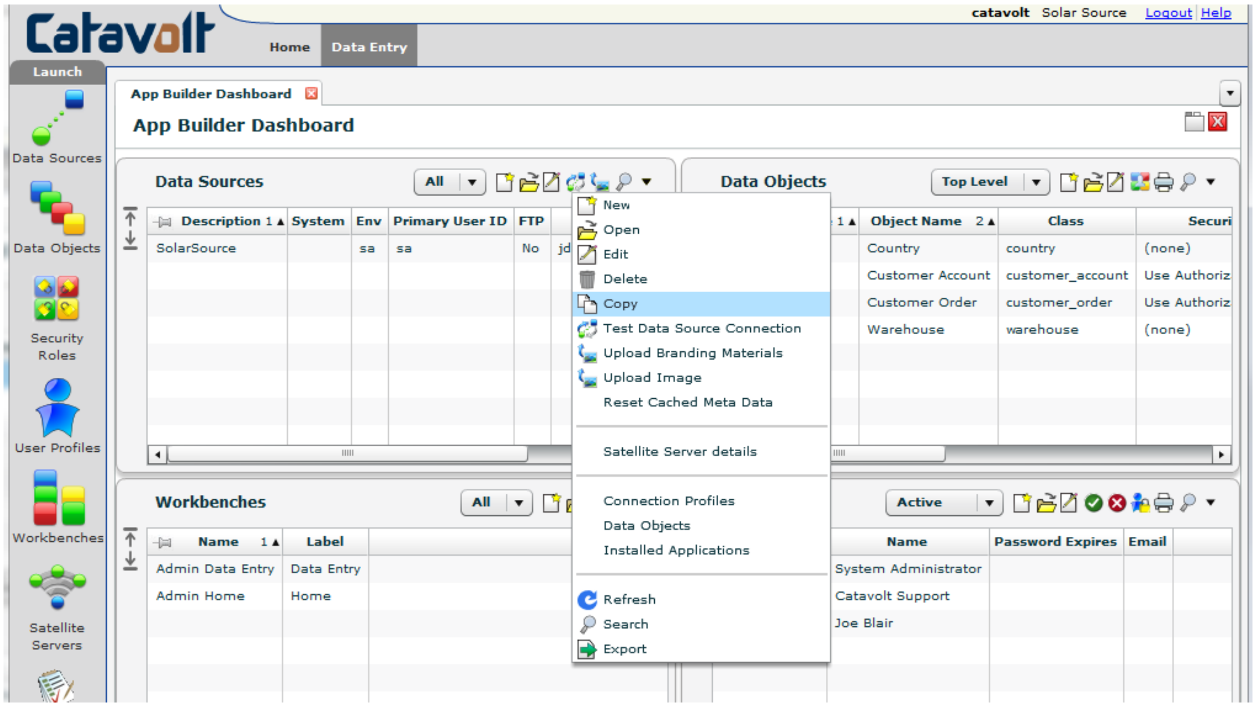
Figure 17: Mobile device Xalt main workbench screen after customer-specific branding has been applied.

Your Hexagon sales representative has more information on how to apply mobile branding to your Xalt application.

Copying a Data Source

A data source can be copied by selecting the Copy menu action from the Data Source toolbar or context menu. There are many reasons to copy a data source. The most common reason is to copy work from a production environment to a development environment to begin a new project. When a data source is copied all of its data objects and connection profiles are copied as well. The new data objects and their corresponding launchers will not be on any workbenches. You must place them on workbenches and give those workbenches to users for the new data objects to be accessible.

When copying a data source, you will be prompted for the name of the new data source and then a new data source will be created with the name that you typed in. After the new data source is created you can then edit it to change its settings such as database URL.



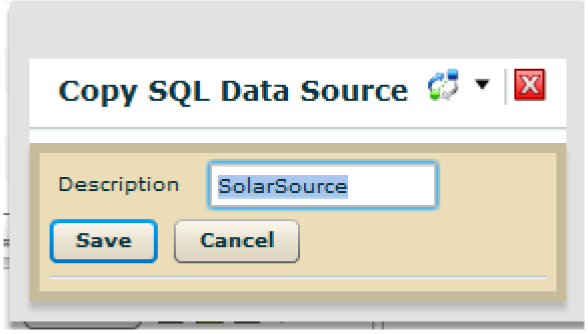
Description 1 ▲	System	Env	Primary User ID	FTP
SolarSource		sa	sa	No

Object Name 2 ▲	Class	Securi
Country	country	(none)
Customer Account	customer_account	Use Authoriz
Customer Order	customer_order	Use Authoriz
Warehouse	warehouse	(none)

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home

Name	Password Expires	Email
System Administrator		
Catavolt Support		
Joe Blair		

Figure 18: Data Source Copy action in context menu.



Description: SolarSource

Figure 19: Data Source Copy Prompt

Chapter 3: Data Objects

Chapter Summary

A Data Object represents an entity from the back-end system, such as a sales order, customer account, item, etc. For most Data Sources, a Data Object represents either a single database table or a group of related database tables. A Data Object allows you to determine the fields from the database table(s) that are visible, how they are presented to the end user, and what other Data Objects can be navigated to.

When an end user brings up Xalt | Mobility, a list of launchers appears down the left side of the window. These launchers represent the available data objects the user can access.

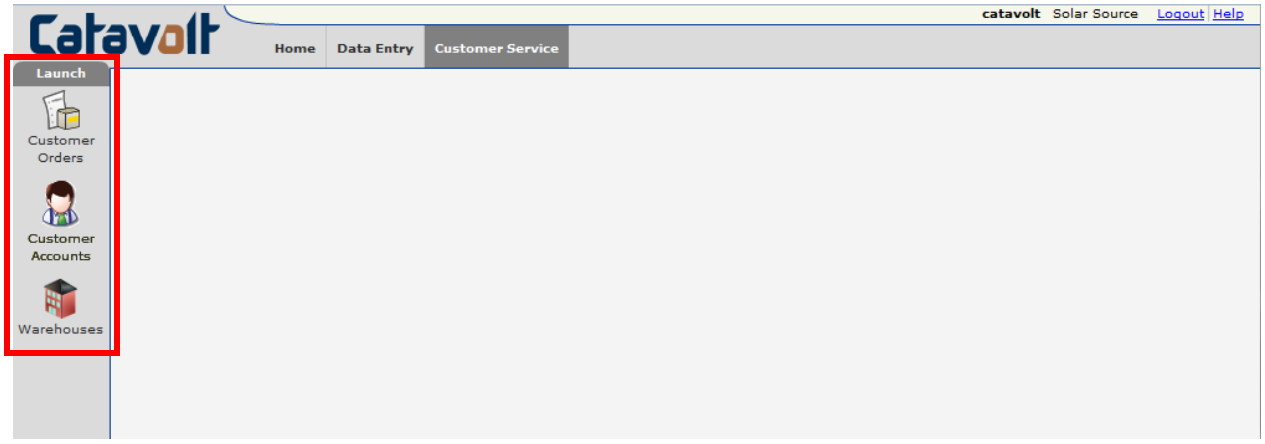
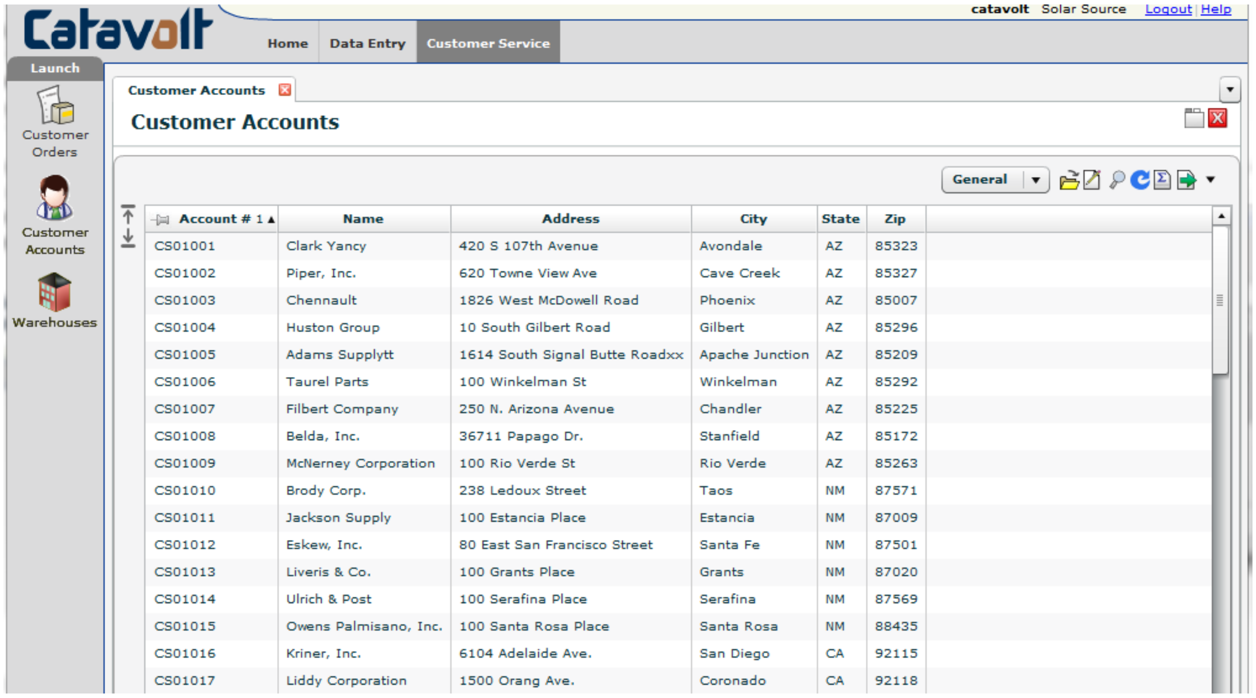


Figure 1: List of launchers in the Xalt window

Clicking on a Launcher will open up the data object. In most cases the launcher will open the data object in query mode. A Query corresponds to a list of records.

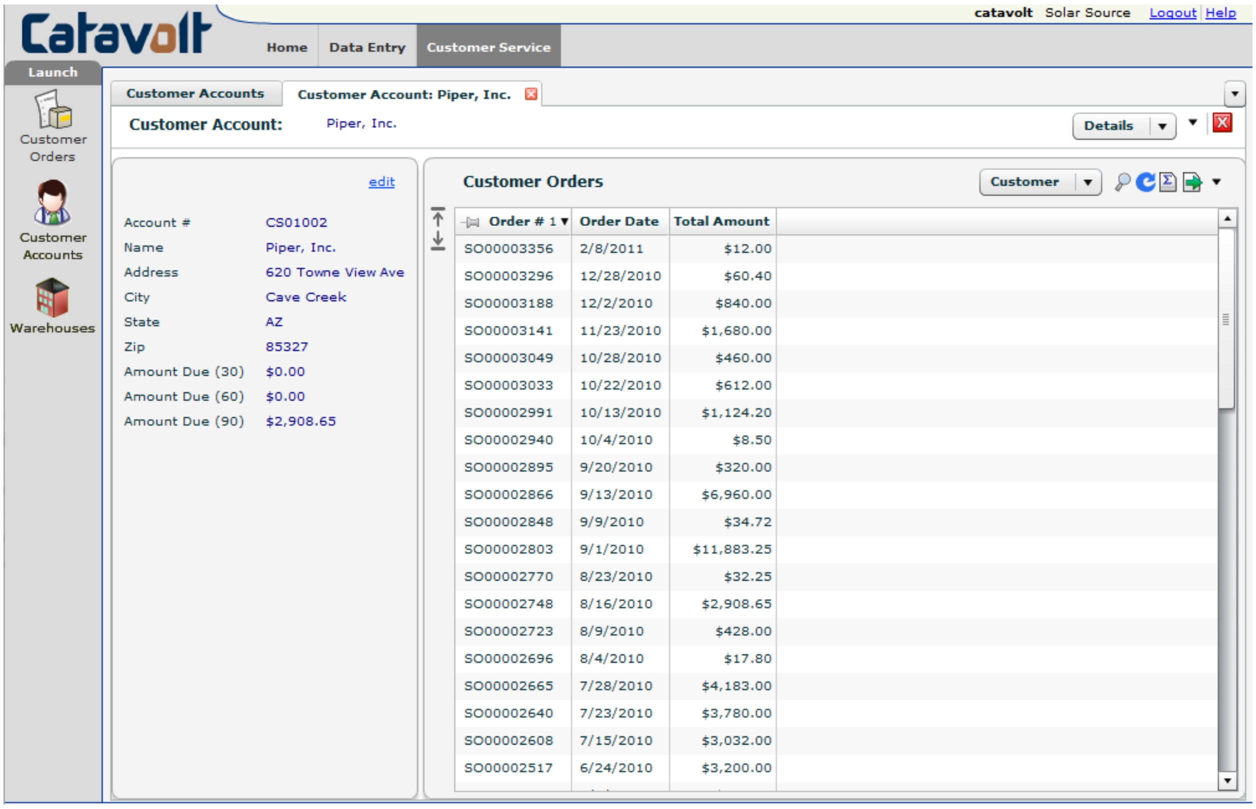


Customer Accounts

Account # 1 ▲	Name	Address	City	State	Zip
CS01001	Clark Yancy	420 S 107th Avenue	Avondale	AZ	85323
CS01002	Piper, Inc.	620 Towne View Ave	Cave Creek	AZ	85327
CS01003	Chennault	1826 West McDowell Road	Phoenix	AZ	85007
CS01004	Huston Group	10 South Gilbert Road	Gilbert	AZ	85296
CS01005	Adams Supplytt	1614 South Signal Butte Roadxx	Apache Junction	AZ	85209
CS01006	Taurel Parts	100 Winkelman St	Winkelman	AZ	85292
CS01007	Filbert Company	250 N. Arizona Avenue	Chandler	AZ	85225
CS01008	Belda, Inc.	36711 Papago Dr.	Stanfield	AZ	85172
CS01009	McNerney Corporation	100 Rio Verde St	Rio Verde	AZ	85263
CS01010	Brody Corp.	238 Ledoux Street	Taos	NM	87571
CS01011	Jackson Supply	100 Estancia Place	Estancia	NM	87009
CS01012	Eskew, Inc.	80 East San Francisco Street	Santa Fe	NM	87501
CS01013	Liveris & Co.	100 Grants Place	Grants	NM	87020
CS01014	Ulrich & Post	100 Serafina Place	Serafina	NM	87569
CS01015	Owens Palmisano, Inc.	100 Santa Rosa Place	Santa Rosa	NM	88435
CS01016	Kriner, Inc.	6104 Adelaide Ave.	San Diego	CA	92115
CS01017	Liddy Corporation	1500 Orang Ave.	Coronado	CA	92118

Figure 2: An example Data Object opened in query mode

Double-clicking or opening up a record in a Query will bring up a Detail for the Data Object. A Detail corresponds to a single record. Note that a Detail may also contain Queries for related Data Objects in one or more of its sections.



Customer Account: Piper, Inc.

Account #	CS01002
Name	Piper, Inc.
Address	620 Towne View Ave
City	Cave Creek
State	AZ
Zip	85327
Amount Due (30)	$0.00
Amount Due (60)	$0.00
Amount Due (90)	$2,908.65

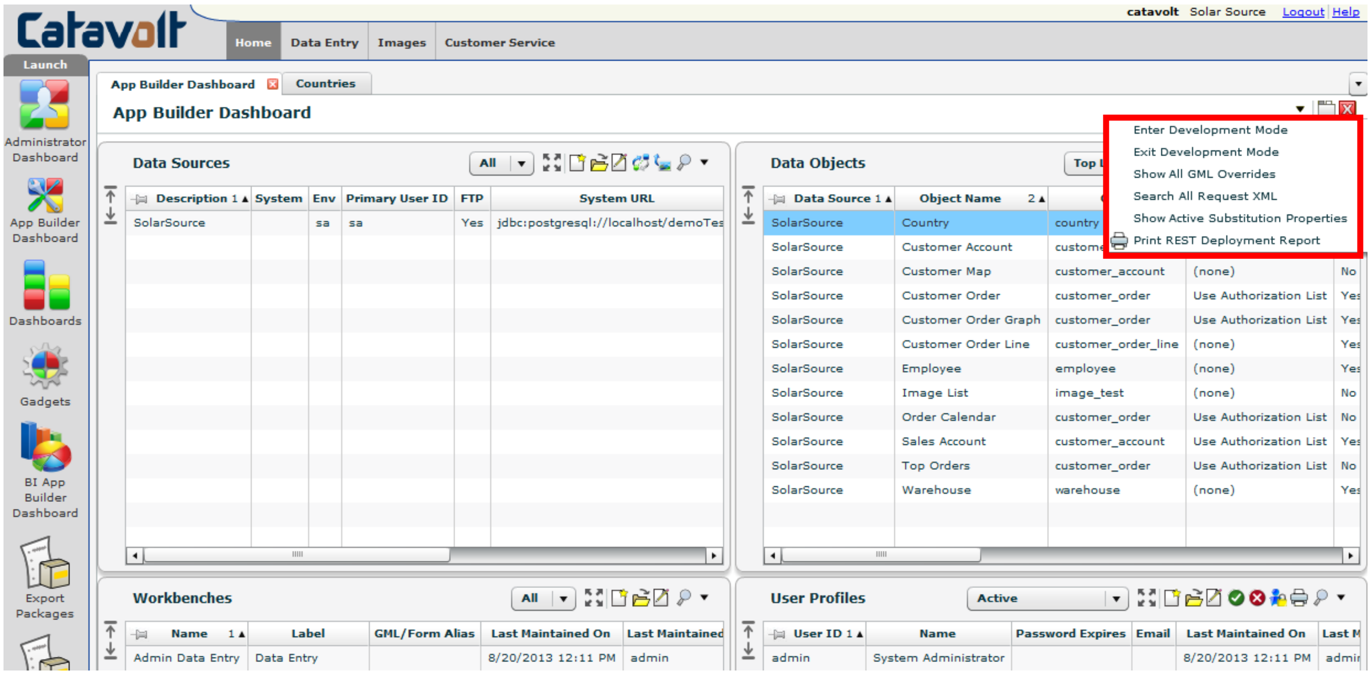
Customer Orders

Order # 1 ▼	Order Date	Total Amount
SO00003356	2/8/2011	$12.00
SO00003296	12/28/2010	$60.40
SO00003188	12/2/2010	$840.00
SO00003141	11/23/2010	$1,680.00
SO00003049	10/28/2010	$460.00
SO00003033	10/22/2010	$612.00
SO00002991	10/13/2010	$1,124.20
SO00002940	10/4/2010	$8.50
SO00002895	9/20/2010	$320.00
SO00002866	9/13/2010	$6,960.00
SO00002848	9/9/2010	$34.72
SO00002803	9/1/2010	$11,883.25
SO00002770	8/23/2010	$32.25
SO00002748	8/16/2010	$2,908.65
SO00002723	8/9/2010	$428.00
SO00002696	8/4/2010	$17.80
SO00002665	7/28/2010	$4,183.00
SO00002640	7/23/2010	$3,780.00
SO00002608	7/15/2010	$3,032.00
SO00002517	6/24/2010	$3,200.00

Figure 3: An example of a Data Object Detail

App Builder Dashboard

When a developer signs on to Xalt | Mobility, they will typically be immediately brought to the App Builder Dashboard. This dashboard is the hub that allows you to create and deploy Xalt applications. From this Dashboard, you can directly access a list of all Data Sources, Data Objects, Workbenches, and User Profiles in the system. In addition, the App Builder Dashboard contains its own menu with some general development options.



App Builder Dashboard

Data Sources

Description 1 ▲	System	Env	Primary User ID	FTP	System URL
SolarSource		sa	sa	Yes	jdbc:postgresql://localhost/demoTes

Data Objects

Data Source 1 ▲	Object Name 2 ▲			
SolarSource	Country	country		
SolarSource	Customer Account	custome		
SolarSource	Customer Map	customer_account	(none)	No
SolarSource	Customer Order	customer_order	Use Authorization List	Yes
SolarSource	Customer Order Graph	customer_order	Use Authorization List	Yes
SolarSource	Customer Order Line	customer_order_line	(none)	Yes
SolarSource	Employee	employee	(none)	Yes
SolarSource	Image List	image_test	(none)	No
SolarSource	Order Calendar	customer_order	Use Authorization List	No
SolarSource	Sales Account	customer_account	Use Authorization List	Yes
SolarSource	Top Orders	customer_order	Use Authorization List	No
SolarSource	Warehouse	warehouse	(none)	Yes

Workbenches

Name 1 ▲	Label	GML/Form Alias	Last Maintained On	Last Maintained
Admin Data Entry	Data Entry		8/20/2013 12:11 PM	admin

User Profiles

User ID 1 ▲	Name	Password Expires	Email	Last Maintained On	Last M
admin	System Administrator			8/20/2013 12:11 PM	admir

Figure 4: The App Builder Dashboard with its menu highlighted

Enter Development Mode

Choosing the Enter Development Mode menu option will temporarily add extra menu options to your Data Objects in runtime to allow you to navigate directly to the corresponding Development Data Object or Query/Detail/Action. In addition, you can bring up a view that will show you the connector gateway calls used to build the data for the window in question.

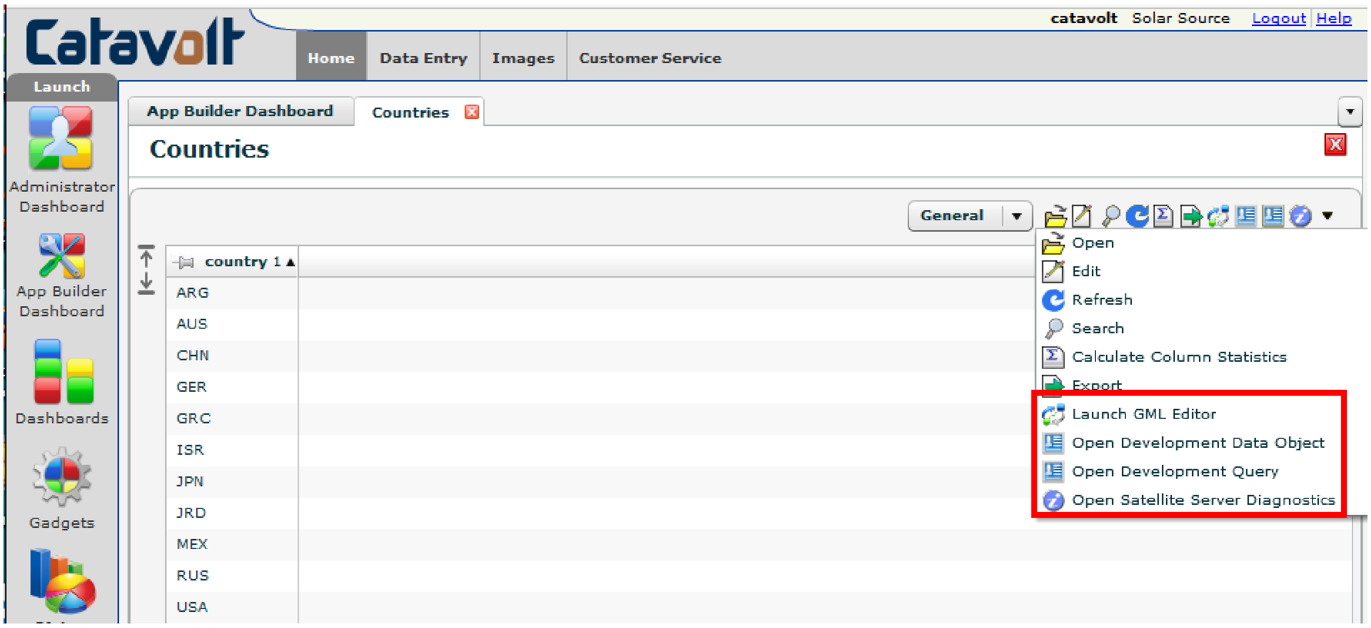
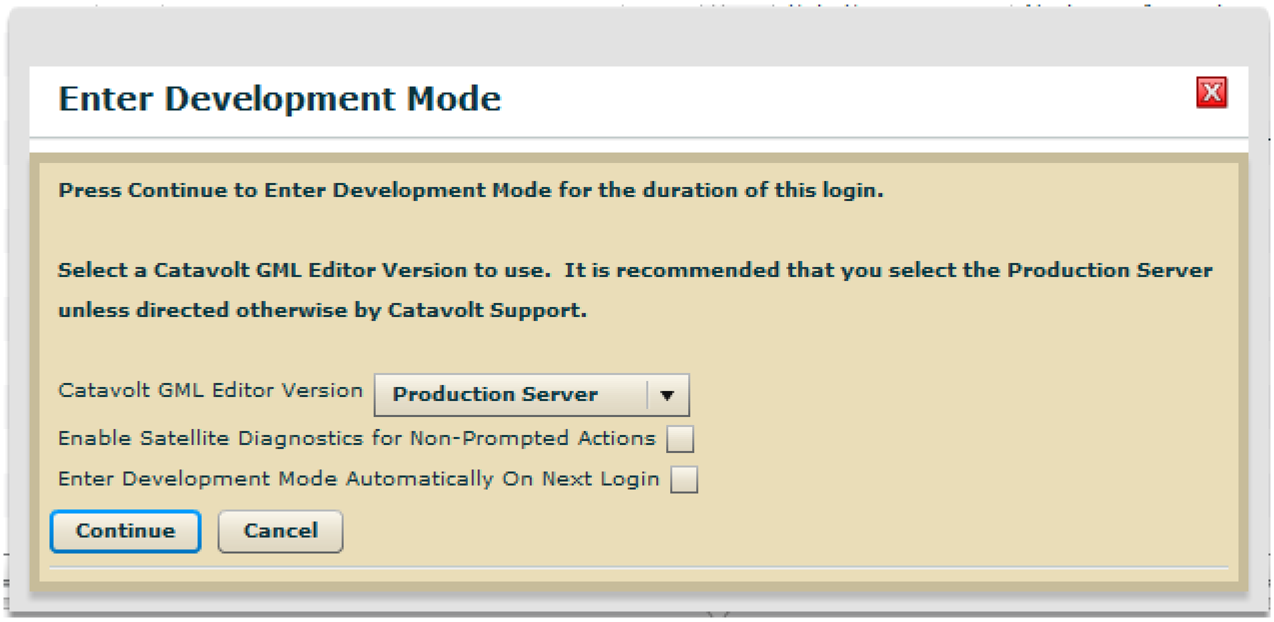


Figure 5: The Countries list with extra Development Mode options

When entering Development Mode, you have the following options



Enter Development Mode

Press Continue to Enter Development Mode for the duration of this login.

Select a Catavolt GML Editor Version to use. It is recommended that you select the Production Server unless directed otherwise by Catavolt Support.

Catavolt GML Editor Version: Production Server ▼

Enable Satellite Diagnostics for Non-Prompted Actions □

Enter Development Mode Automatically On Next Login □

Figure 6: The Enter Development Mode details window

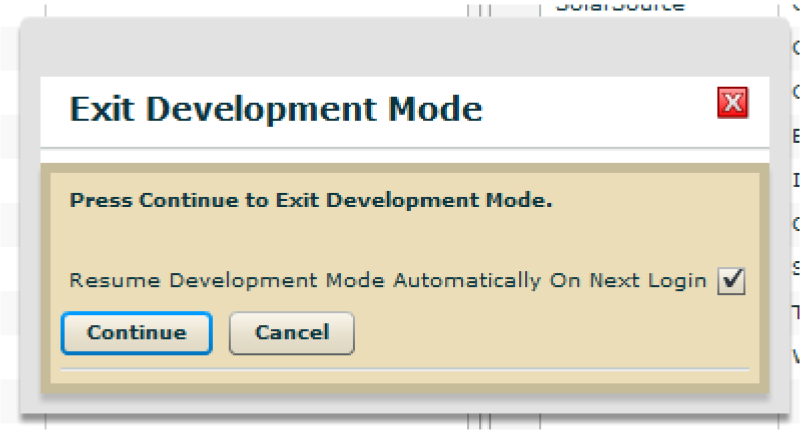
Hexagon GML Editor Version - If you are licensed to use GML, this menu option will also allow you to edit GML in real time using live data from your application. Your Hexagon sales representative has more information about using GML in your application.

Enable Satellite Diagnostics for Non-Prompted Actions will put up a debug prompt when running any Actions that do not have an Action Prompt. This will provide you with the ability to access the special menu actions in Development Mode for actions that do not have prompts.

Enter Development Mode Automatically On Next Login will automatically put you Development Mode each time you login. If unchecked, Development Mode will turn off when you logoff.

Exit Development Mode

Choosing the Exit Development Mode menu option will turn Development Mode off for your session.


Exit Development Mode

Press Continue to Exit Development Mode.

Resume Development Mode Automatically On Next Login ☑

Figure 7: The Exit Development Mode details window

Resume Development Mode Automatically On Next Login will automatically put you back into Development Mode once you logoff and back on.

Show All GML Overrides

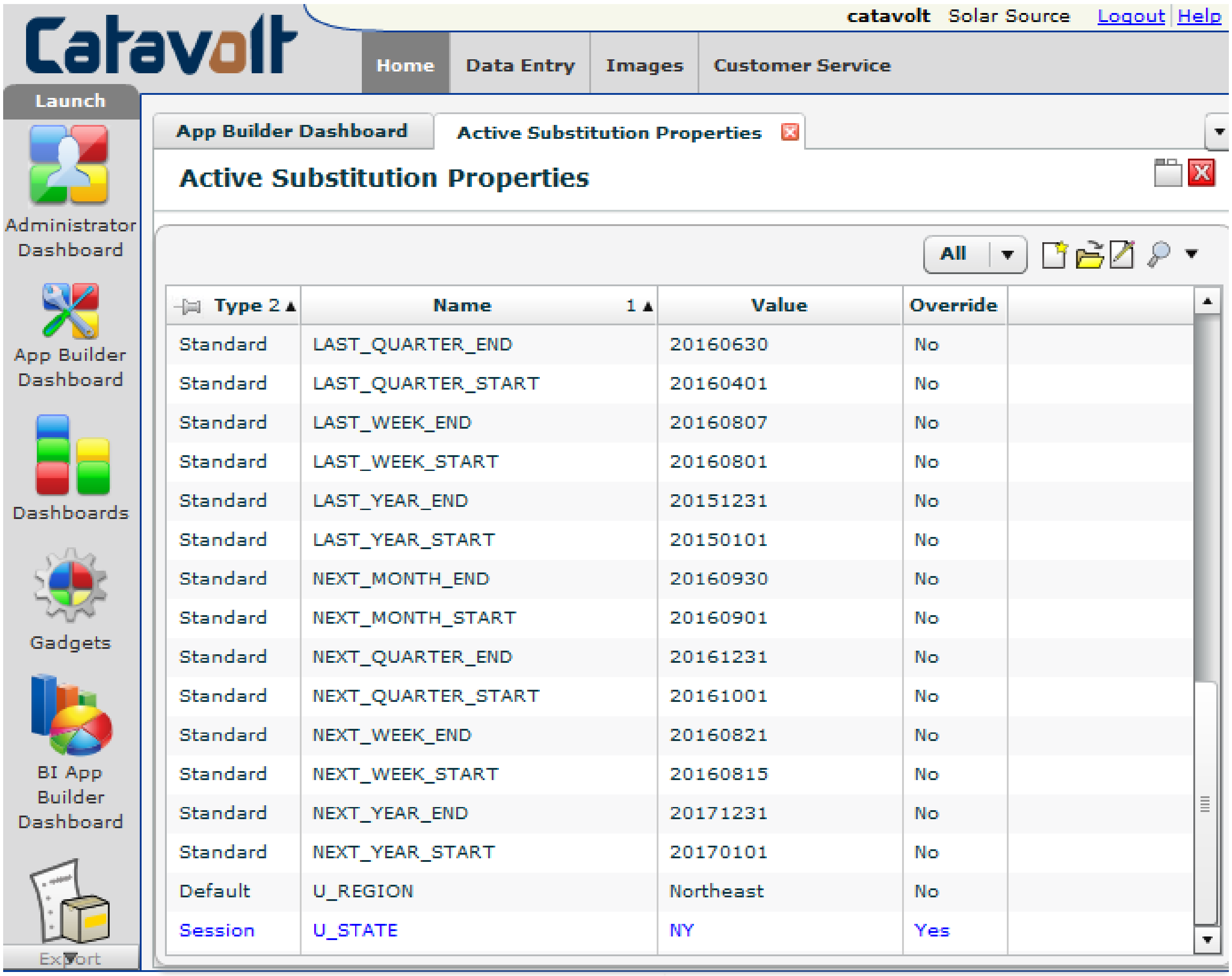
Choosing the Show All GML Overrides menu option will open a window that will display all objects that contain GML Overrides. Double-clicking on any record in this list will take you directly to the GML Override and allow you to edit the data. In addition, there is a "Search All GML" menu option that will allow you to search all GML Overrides for a particular string. Your Hexagon sales representative has more information about using GML in your application.

Search All Request XML

Choosing the Search All Request XML will allow you to search Request XML across all Actions for a particular string. This can be used to locate actions where a specific Stored Procedure is called, for example.

Show Active Substitution Properties

Choosing the Show Active Substitution Properties menu option will open a window that will display all current Substitution Properties that are in effect for the current logged-in user. This is helpful during application development to see what values are currently in effect. The window will show you the name of the Property and its current value. In addition, it will show you the type of value (Standard, Default, User, or Session) and whether the property has been overridden.

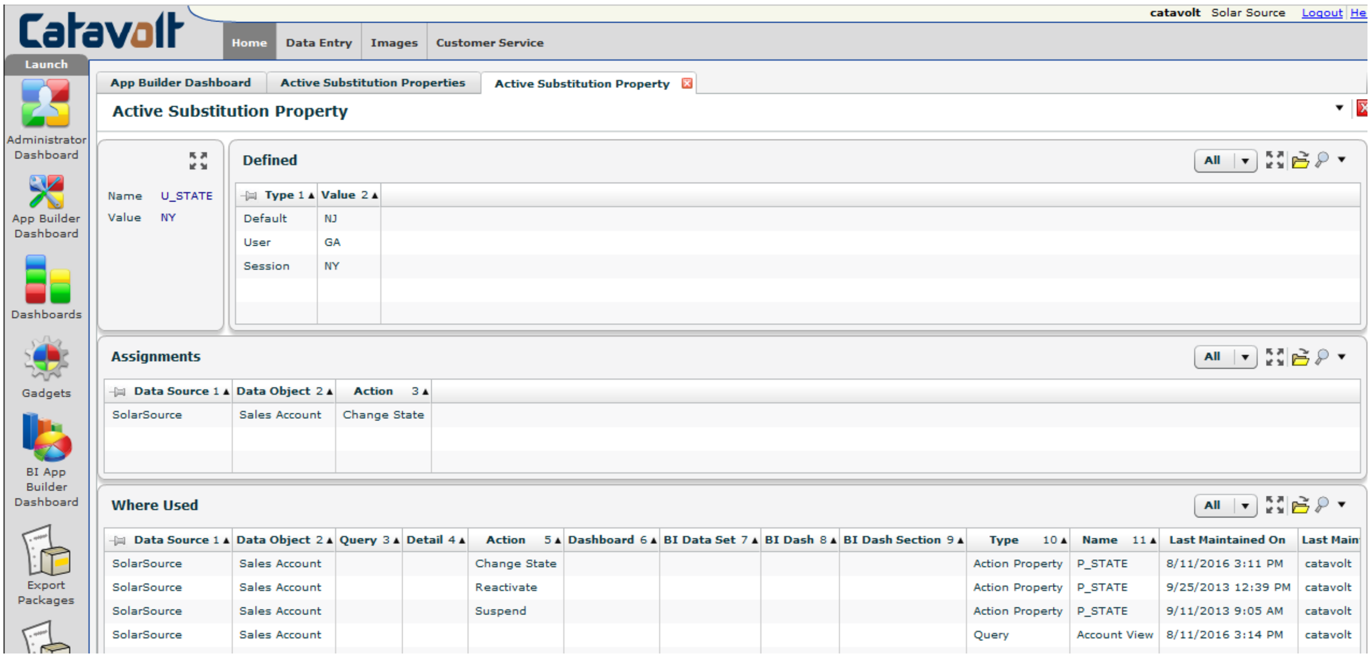


Active Substitution Properties

Type 2 ▲	Name 1 ▲	Value	Override
Standard	LAST_QUARTER_END	20160630	No
Standard	LAST_QUARTER_START	20160401	No
Standard	LAST_WEEK_END	20160807	No
Standard	LAST_WEEK_START	20160801	No
Standard	LAST_YEAR_END	20151231	No
Standard	LAST_YEAR_START	20150101	No
Standard	NEXT_MONTH_END	20160930	No
Standard	NEXT_MONTH_START	20160901	No
Standard	NEXT_QUARTER_END	20161231	No
Standard	NEXT_QUARTER_START	20161001	No
Standard	NEXT_WEEK_END	20160821	No
Standard	NEXT_WEEK_START	20160815	No
Standard	NEXT_YEAR_END	20171231	No
Standard	NEXT_YEAR_START	20170101	No
Default	U_REGION	Northeast	No
Session	U_STATE	NY	Yes

Figure 8: Active Substitution Properties list

Double-clicking the record will bring up more details.



Active Substitution Property

Name U_STATE
Value NY

Defined

Type 1	Value 2
Default	NJ
User	GA
Session	NY

Assignments

Data Source 1	Data Object 2	Action 3
SolarSource	Sales Account	Change State

Where Used

Data Source 1	Data Object 2	Query 3	Detail 4	Action 5	Dashboard 6	BI Data Set 7	BI Dash 8	BI Dash Section 9	Type 10	Name 11	Last Maintained On	Last Main
SolarSource	Sales Account			Change State					Action Property	P_STATE	8/11/2016 3:11 PM	catavolt
SolarSource	Sales Account			Reactivate					Action Property	P_STATE	9/25/2013 12:39 PM	catavolt
SolarSource	Sales Account			Suspend					Action Property	P_STATE	9/11/2013 9:05 AM	catavolt
SolarSource	Sales Account								Query	Account View	8/11/2016 3:14 PM	catavolt

Figure 9: Active Substitution Property details view

The Details has the following sub-sections:

Defined

Defined shows the current levels where this property is defined. You may have a record defined at the Default (Default Property), User (User Profile Property) and/or Session (Session Property) levels. Double-clicking a Default record will bring up details for the Default Property. Double-clicking a User record will allow you to edit the User Profile Property value.

Assignments

Assignments shows Action Request XML sections where this value is set via a SetSessionProperty, SetUserProperty, or SetDefaultProperty action. You can double-click the record to bring up the referenced Action.

Where Used

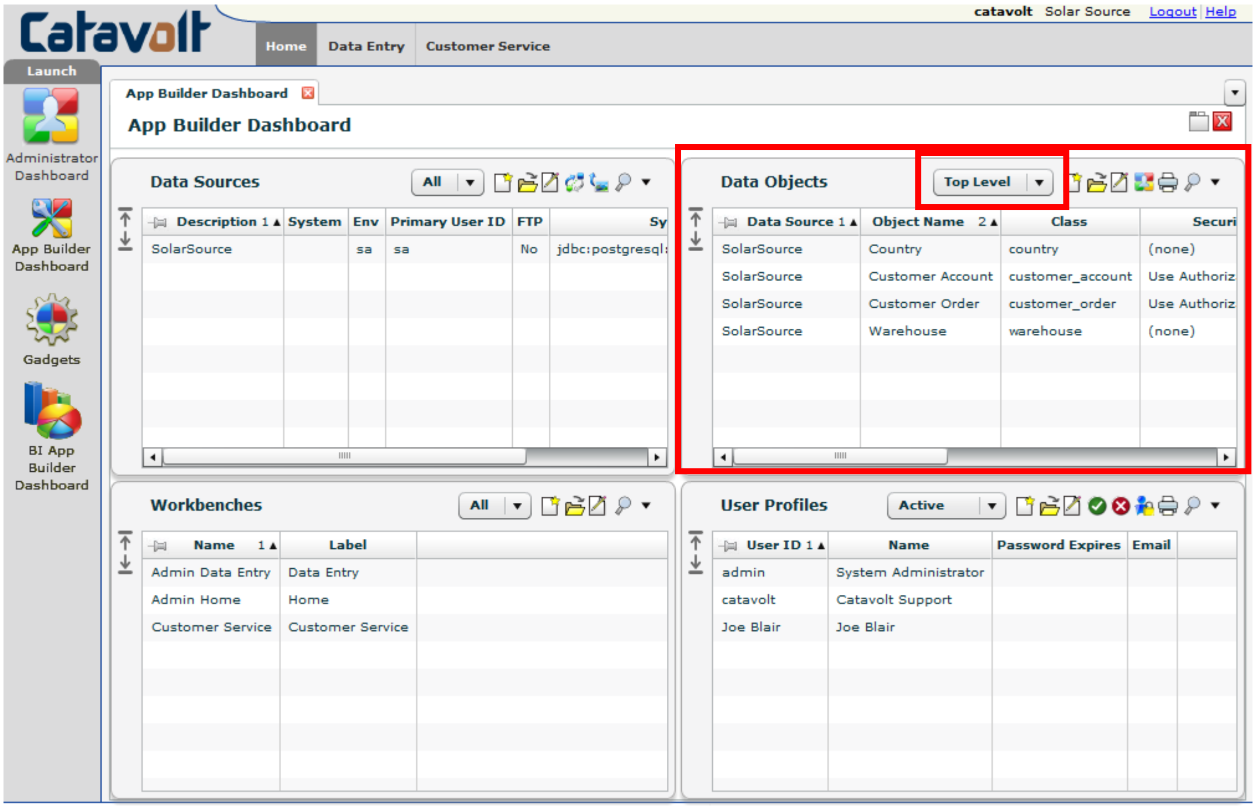
Where Used shows all of the objects that are currently using this property. You can double-click the record to bring up the referenced object.

Print REST Deployment Report

Choosing the Print REST Deployment Report menu option will open a window that will display a complete listing of what Data Objects you have deployed in the Hexagon REST API. Please see the **Hexagon REST API Guide** for more information on the Hexagon REST API.

Accessing Data Objects

There are 2 ways for a developer to see and configure what Data Objects are available to be used in Xalt | Mobility. The Data Objects section of the App Builder Dashboard shows you a list of all Data Objects defined in Xalt sorted by Data Source and Object Name. There are two views for Data Objects. The Top Level view only shows Data Objects that have Launchers and can be put directly on the main Xalt window. The All view shows every defined Data Object.



Data Sources

Description 1▲	System	Env	Primary User ID	FTP	Sy
SolarSource		sa	sa	No	jdbc:postgresql:

Data Objects

Data Source 1▲	Object Name 2▲	Class	Securi
SolarSource	Country	country	(none)
SolarSource	Customer Account	customer_account	Use Authoriz
SolarSource	Customer Order	customer_order	Use Authoriz
SolarSource	Warehouse	warehouse	(none)

Workbenches

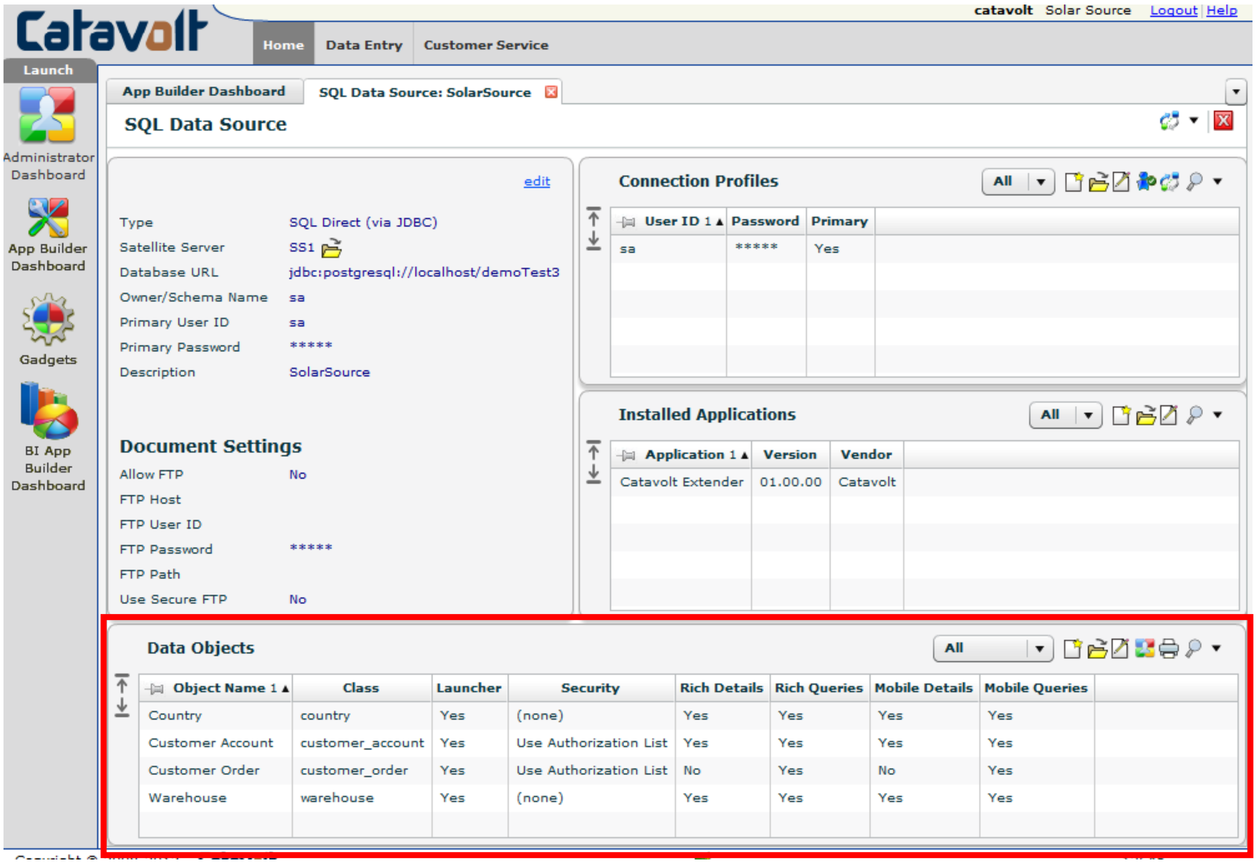
Name 1▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

User Profiles

User ID 1▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Figure 10: The App Builder Dashboard with the Data Objects list highlighted and the available views drop down also highlighted.

When viewing a Data Source, the Data Objects section shows a list of all Data Objects defined for that particular Data Source. As above, both the Top Level and All views are available.



SQL Data Source

Type	SQL Direct (via JDBC)
Satellite Server	SS1
Database URL	jdbc:postgresql://localhost/demoTest3
Owner/Schema Name	sa
Primary User ID	sa
Primary Password	*****
Description	SolarSource

Document Settings

Allow FTP	No
FTP Host	
FTP User ID	
FTP Password	*****
FTP Path	
Use Secure FTP	No

Connection Profiles

User ID 1 ▲	Password	Primary
sa	*****	Yes

Installed Applications

Application 1 ▲	Version	Vendor
Catavolt Extender	01.00.00	Catavolt

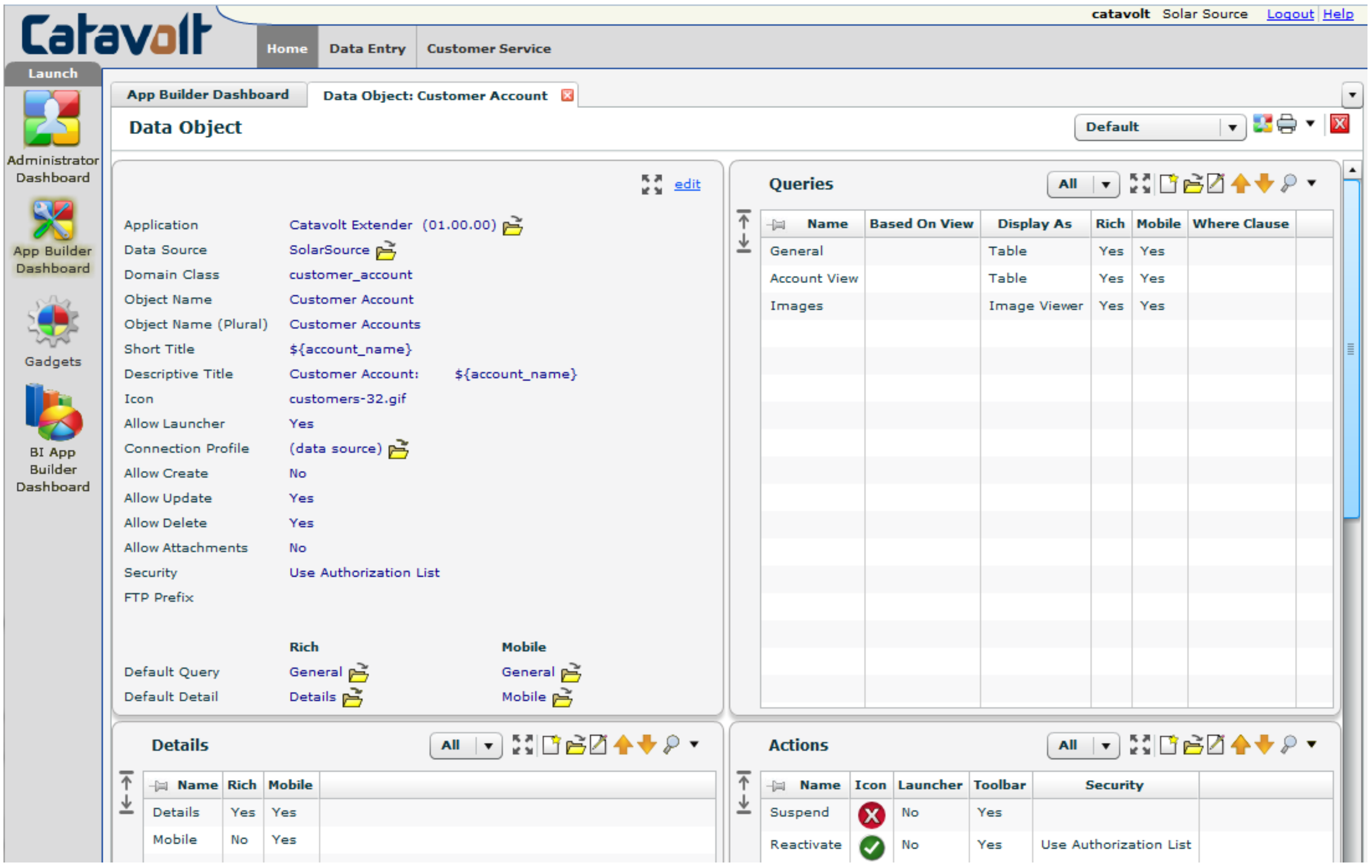
Data Objects

Object Name 1 ▲	Class	Launcher	Security	Rich Details	Rich Queries	Mobile Details	Mobile Queries
Country	country	Yes	(none)	Yes	Yes	Yes	Yes
Customer Account	customer_account	Yes	Use Authorization List	Yes	Yes	Yes	Yes
Customer Order	customer_order	Yes	Use Authorization List	No	Yes	No	Yes
Warehouse	warehouse	Yes	(none)	Yes	Yes	Yes	Yes

Figure 11: Data Source details view with the Data Objects query section highlighted.

Data Object Components

When you display the definition for a Data Object, there are 3 Details that you can select from. The Default Detail shows you all the information about the Data Object. In addition, you can see what Queries (list) and Details (object) are available for the Data Object, as well as what (menu and toolbar) Actions have been defined for the Data Object, and what Permissions (if any) are attached to the Data Object.



Application	Catavolt Extender (01.00.00)	
Data Source	SolarSource	
Domain Class	customer_account	
Object Name	Customer Account	
Object Name (Plural)	Customer Accounts	
Short Title	${account_name}	
Descriptive Title	Customer Account: ${account_name}	
Icon	customers-32.gif	
Allow Launcher	Yes	
Connection Profile	(data source)	
Allow Create	No	
Allow Update	Yes	
Allow Delete	Yes	
Allow Attachments	No	
Security	Use Authorization List	
FTP Prefix		
	Rich	Mobile
Default Query	General	General
Default Detail	Details	Mobile

Queries

Name	Based On View	Display As	Rich	Mobile	Where Clause
General		Table	Yes	Yes	
Account View		Table	Yes	Yes	
Images		Image Viewer	Yes	Yes	

Details

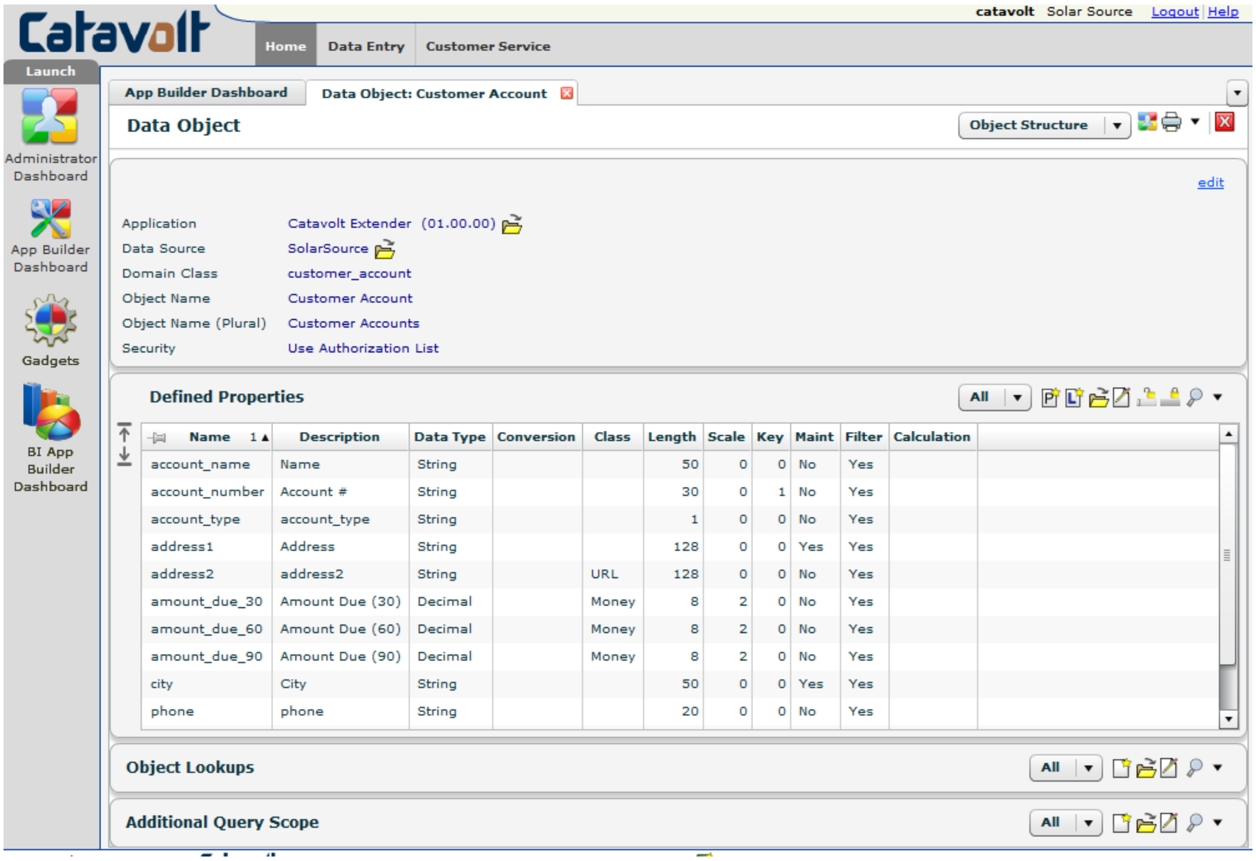
Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

Actions

Name	Icon	Launcher	Toolbar	Security
Suspend	✗	No	Yes	
Reactivate	✓	No	Yes	Use Authorization List

Figure 12: Data Object 'Default' details

The Object Structure Detail shows you what Properties (fields) have been defined for this Data Object. The Object Lookups section allows you to define lookup actions for some properties that allow you to pick a property's value from a list instead of having to manually type it in. The Additional Query Scope section allows you to add properties from related database tables as part of this Data Object (for example, being able to show fields from the Customer table on the Customer Order Data Object).



Data Object

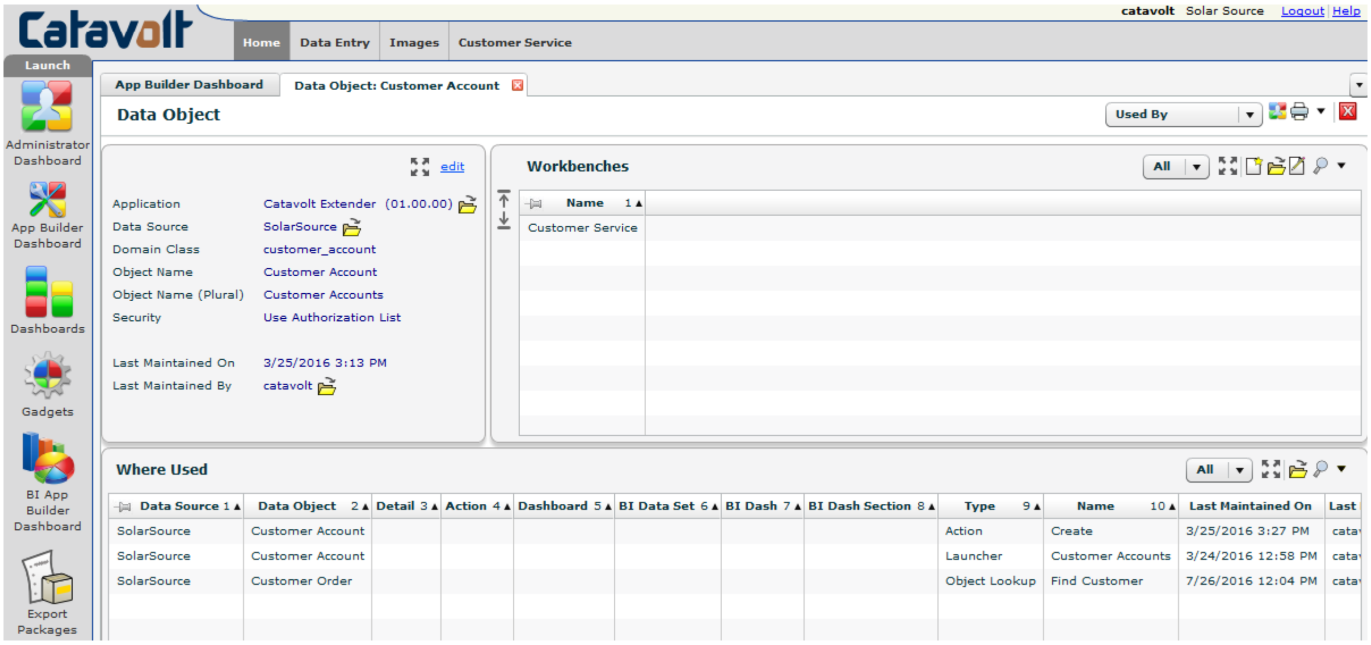
Application Catavolt Extender (01.00.00)
Data Source SolarSource
Domain Class customer_account
Object Name Customer Account
Object Name (Plural) Customer Accounts
Security Use Authorization List

Defined Properties

Name 1 ▲	Description	Data Type	Conversion	Class	Length	Scale	Key	Maint	Filter	Calculation
account_name	Name	String			50	0	0	No	Yes	
account_number	Account #	String			30	0	1	No	Yes	
account_type	account_type	String			1	0	0	No	Yes	
address1	Address	String			128	0	0	Yes	Yes	
address2	address2	String		URL	128	0	0	No	Yes	
amount_due_30	Amount Due (30)	Decimal		Money	8	2	0	No	Yes	
amount_due_60	Amount Due (60)	Decimal		Money	8	2	0	No	Yes	
amount_due_90	Amount Due (90)	Decimal		Money	8	2	0	No	Yes	
city	City	String			50	0	0	Yes	Yes	
phone	phone	String			20	0	0	No	Yes	

Figure 13: Data Object 'Object Structure' details

The Used By Detail shows a list of all Workbenches that currently have this Data Object on them as well as a Where Used list of objects that are currently using this Data Object.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Security	Use Authorization List
Last Maintained On	3/25/2016 3:13 PM
Last Maintained By	catavolt

Workbenches

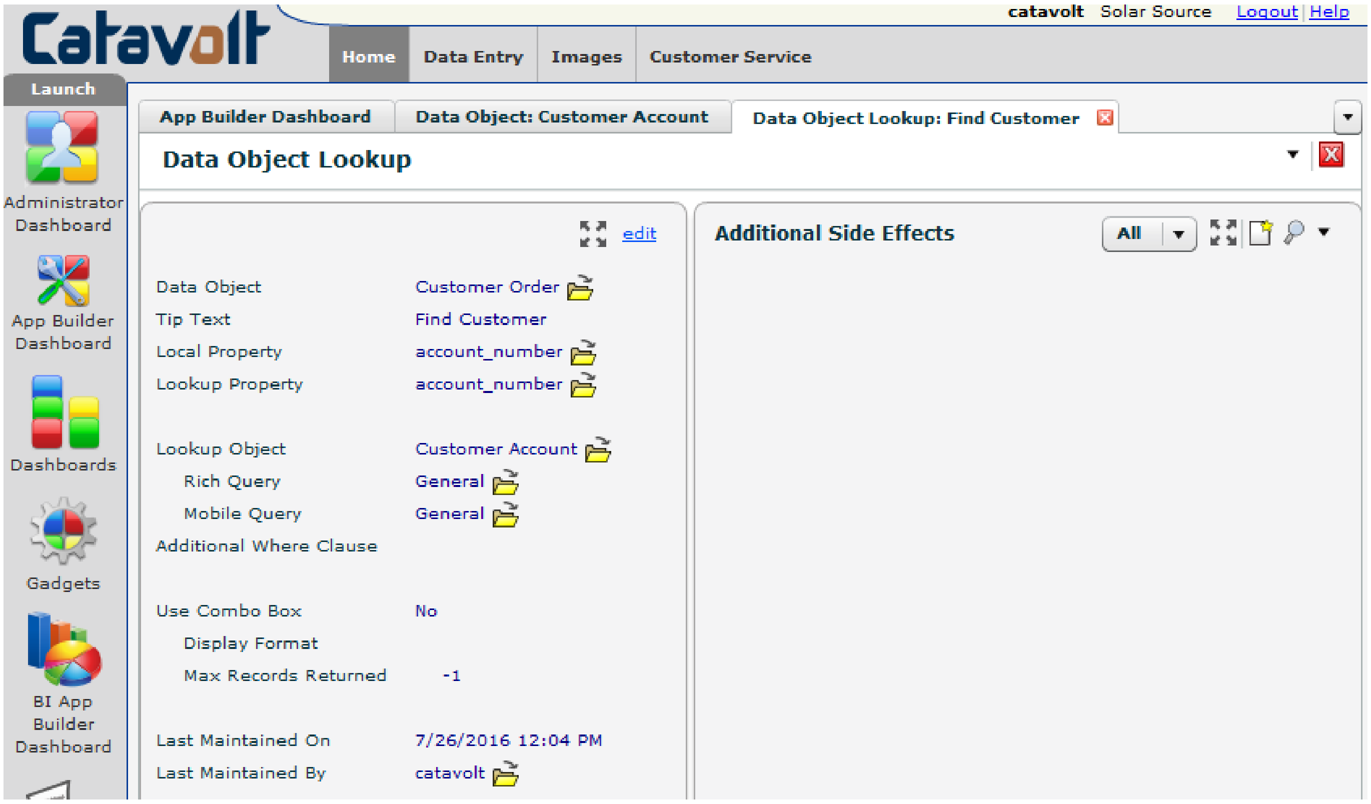
Name 1
Customer Service

Where Used

Data Source 1	Data Object 2	Detail 3	Action 4	Dashboard 5	BI Data Set 6	BI Dash 7	BI Dash Section 8	Type 9	Name 10	Last Maintained On	Last
SolarSource	Customer Account							Action	Create	3/25/2016 3:27 PM	cata
SolarSource	Customer Account							Launcher	Customer Accounts	3/24/2016 12:58 PM	cata
SolarSource	Customer Order							Object Lookup	Find Customer	7/26/2016 12:04 PM	cata

Figure 14: Data Object 'Used By' details

Opening a Where Used record will take you directly to the object using the Data Object:



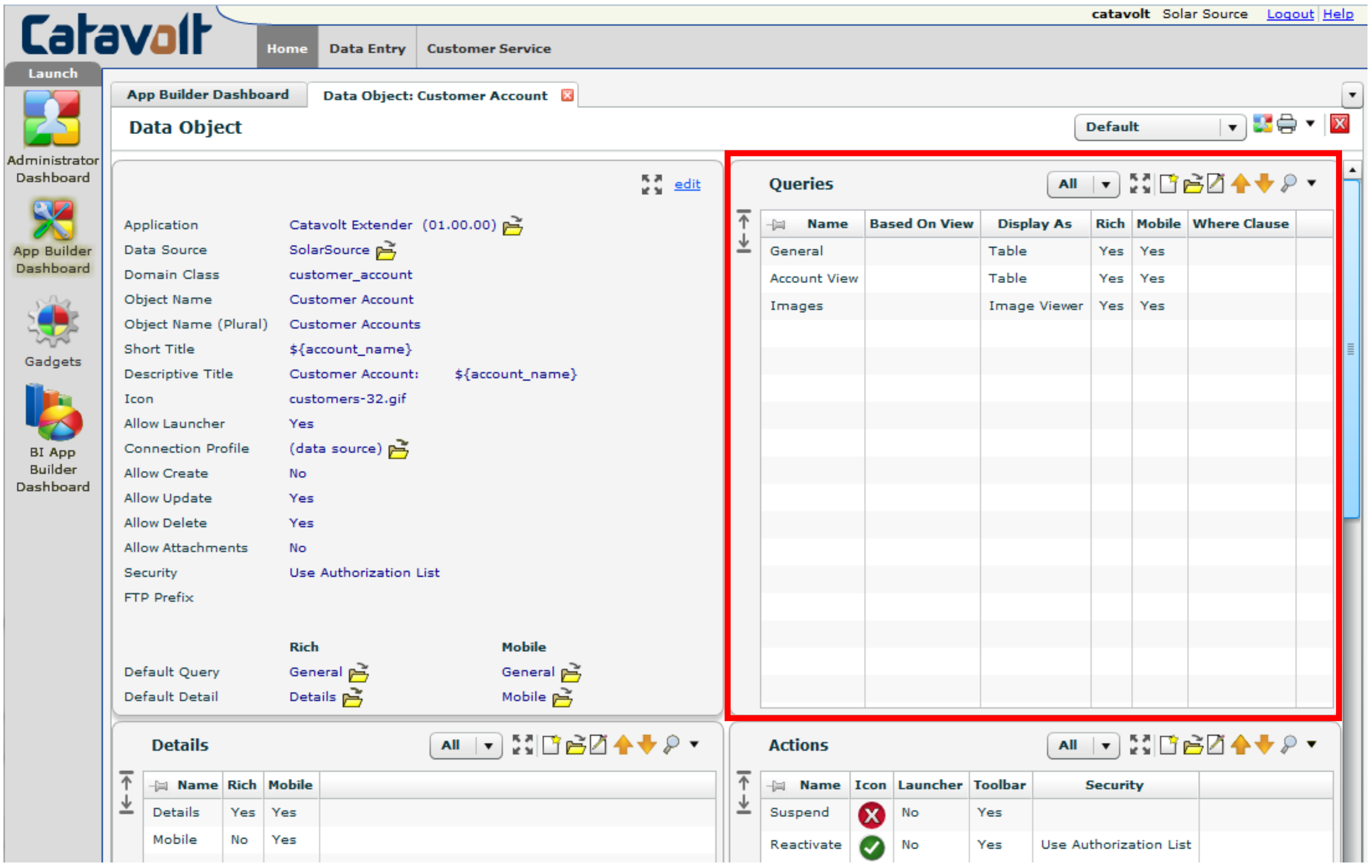
Data Object Lookup

Data Object	Customer Order
Tip Text	Find Customer
Local Property	account_number
Lookup Property	account_number
Lookup Object	Customer Account
Rich Query	General
Mobile Query	General
Additional Where Clause	
Use Combo Box	No
Display Format	
Max Records Returned	-1
Last Maintained On	7/26/2016 12:04 PM
Last Maintained By	catavolt

Figure 15: Data Object Where Used

Queries

A Query represents a list of records. At its core, a Query is made up of 3 parts: a list of what properties (fields, columns) to show, what order to show them in (sort), and (optionally) extra filtering logic to determine which subset of records to show.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Short Title	${account_name}
Descriptive Title	Customer Account: ${account_name}
Icon	customers-32.gif
Allow Launcher	Yes
Connection Profile	(data source)
Allow Create	No
Allow Update	Yes
Allow Delete	Yes
Allow Attachments	No
Security	Use Authorization List
FTP Prefix	

	Rich	Mobile
Default Query	General	General
Default Detail	Details	Mobile

Queries

Name	Based On View	Display As	Rich	Mobile	Where Clause
General		Table	Yes	Yes	
Account View		Table	Yes	Yes	
Images		Image Viewer	Yes	Yes	

Details

Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

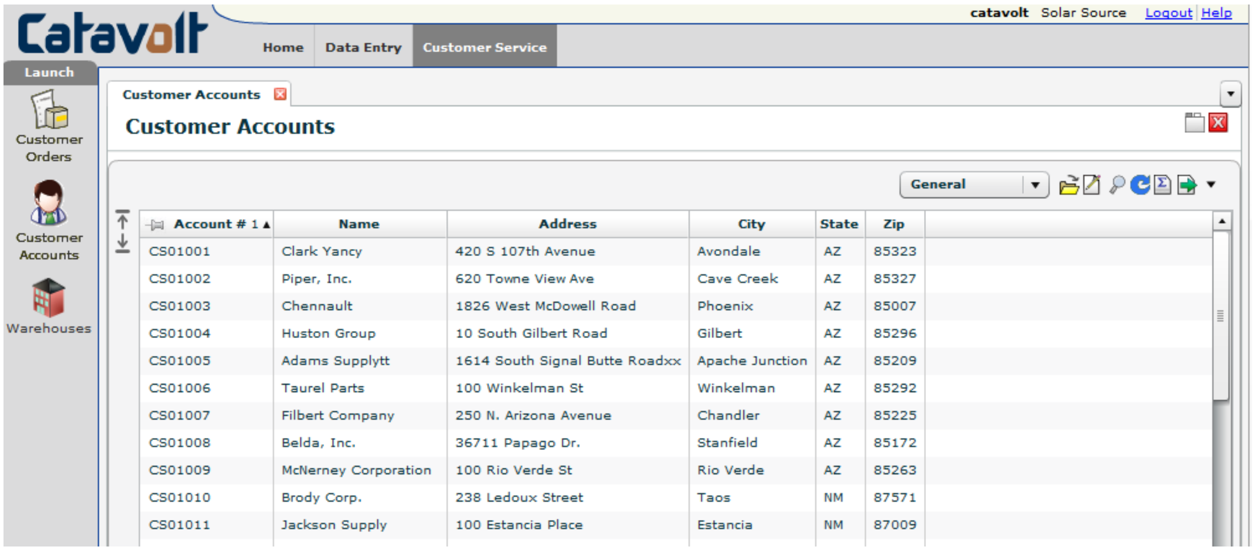
Actions

Name	Icon	Launcher	Toolbar	Security
Suspend	✗	No	Yes	
Reactivate	✓	No	Yes	Use Authorization List

Figure 16: The Data Object 'Default' details with the Queries section highlighted.

In Xalt, you are allowed to define multiple Queries for a particular Data Object. Each Query can provide a different view over the Data Object. This allows you to logically group the columns, sort, and record filtering into named combinations. The list of available queries is selectable by the user via a drop down list that is located next to the toolbar.

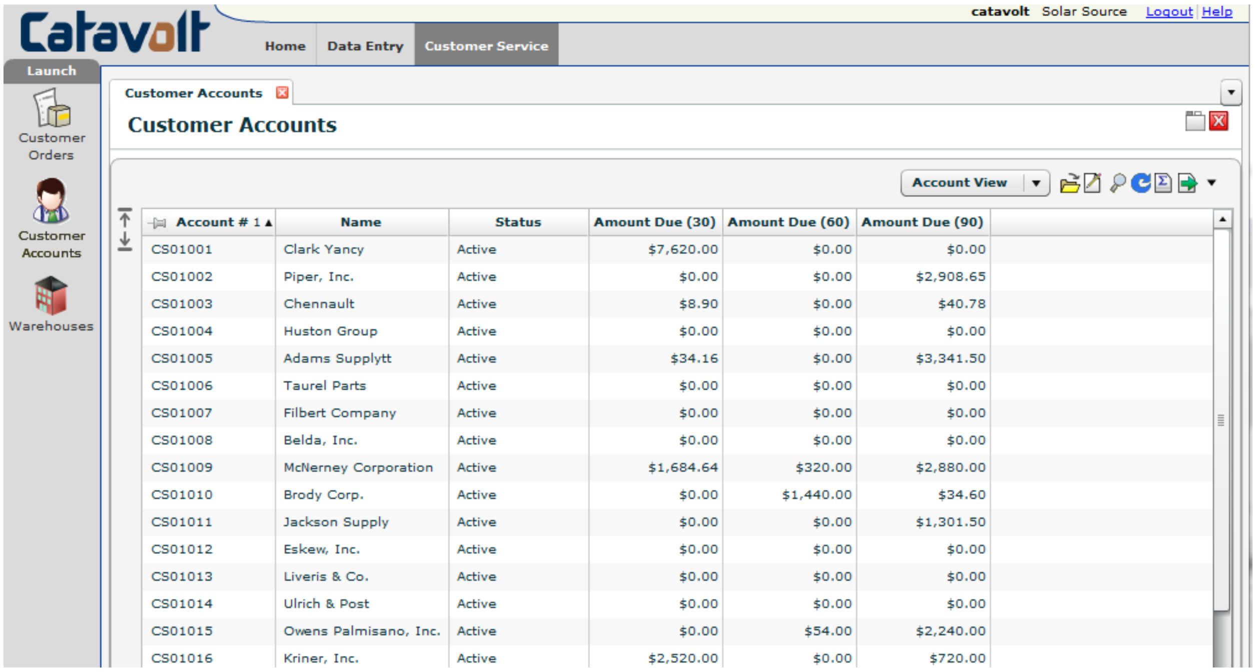
For example, the General Query may show you address information about a Customer:



Account # 1 ▲	Name	Address	City	State	Zip
CS01001	Clark Yancy	420 S 107th Avenue	Avondale	AZ	85323
CS01002	Piper, Inc.	620 Towne View Ave	Cave Creek	AZ	85327
CS01003	Chennault	1826 West McDowell Road	Phoenix	AZ	85007
CS01004	Huston Group	10 South Gilbert Road	Gilbert	AZ	85296
CS01005	Adams Supplytt	1614 South Signal Butte Roadxx	Apache Junction	AZ	85209
CS01006	Taurel Parts	100 Winkelman St	Winkelman	AZ	85292
CS01007	Filbert Company	250 N. Arizona Avenue	Chandler	AZ	85225
CS01008	Belda, Inc.	36711 Papago Dr.	Stanfield	AZ	85172
CS01009	McNerney Corporation	100 Rio Verde St	Rio Verde	AZ	85263
CS01010	Brody Corp.	238 Ledoux Street	Taos	NM	87571
CS01011	Jackson Supply	100 Estancia Place	Estancia	NM	87009

Figure 17: An example Customer data object showing a query named General.

While the Account View query may only show account status information:



Account # 1 ▲	Name	Status	Amount Due (30)	Amount Due (60)	Amount Due (90)
CS01001	Clark Yancy	Active	$7,620.00	$0.00	$0.00
CS01002	Piper, Inc.	Active	$0.00	$0.00	$2,908.65
CS01003	Chennault	Active	$8.90	$0.00	$40.78
CS01004	Huston Group	Active	$0.00	$0.00	$0.00
CS01005	Adams Supplytt	Active	$34.16	$0.00	$3,341.50
CS01006	Taurel Parts	Active	$0.00	$0.00	$0.00
CS01007	Filbert Company	Active	$0.00	$0.00	$0.00
CS01008	Belda, Inc.	Active	$0.00	$0.00	$0.00
CS01009	McNerney Corporation	Active	$1,684.64	$320.00	$2,880.00
CS01010	Brody Corp.	Active	$0.00	$1,440.00	$34.60
CS01011	Jackson Supply	Active	$0.00	$0.00	$1,301.50
CS01012	Eskew, Inc.	Active	$0.00	$0.00	$0.00
CS01013	Liveris & Co.	Active	$0.00	$0.00	$0.00
CS01014	Ulrich & Post	Active	$0.00	$0.00	$0.00
CS01015	Owens Palmisano, Inc.	Active	$0.00	$54.00	$2,240.00
CS01016	Kriner, Inc.	Active	$2,520.00	$0.00	$720.00

Figure 18: An example Customer data object showing a query named 'Account View'

Other Queries may be setup to represent Data Objects using a graph instead of a list:

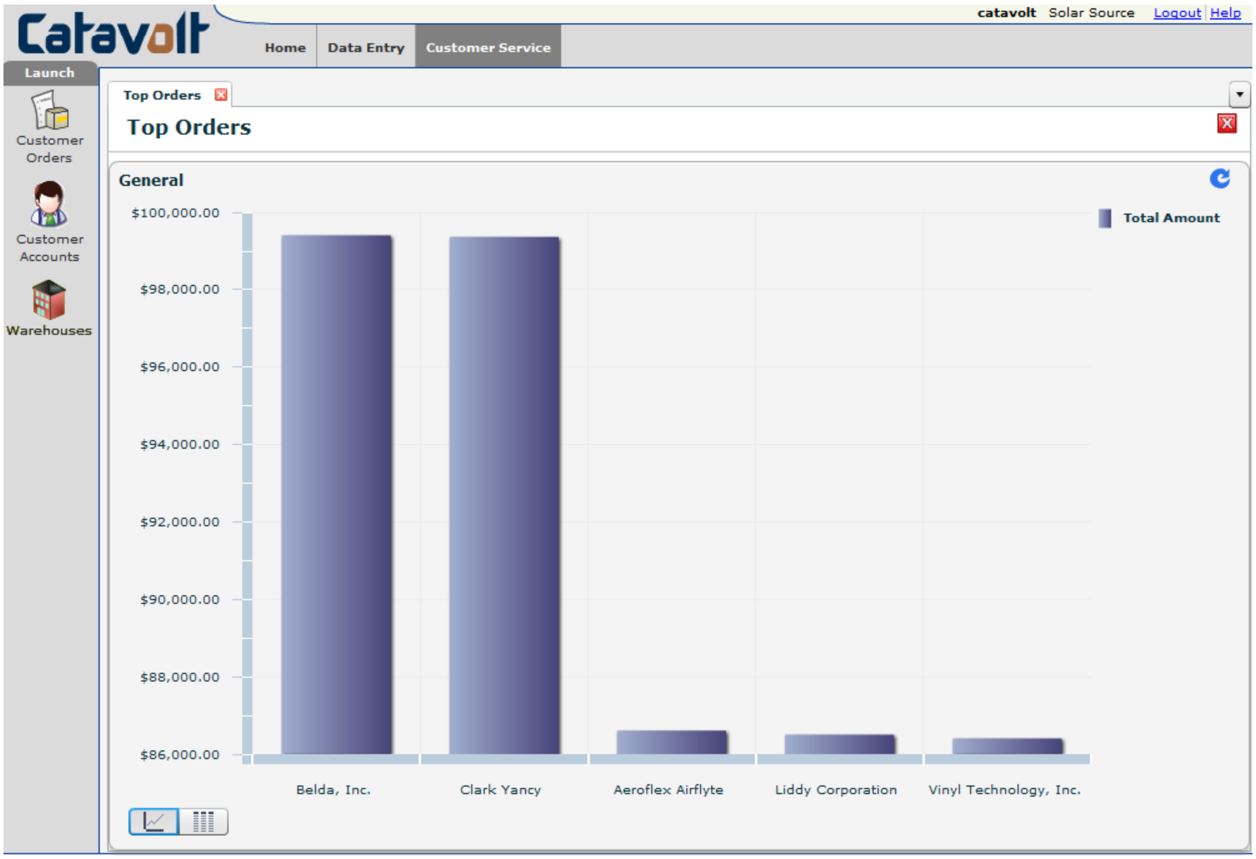
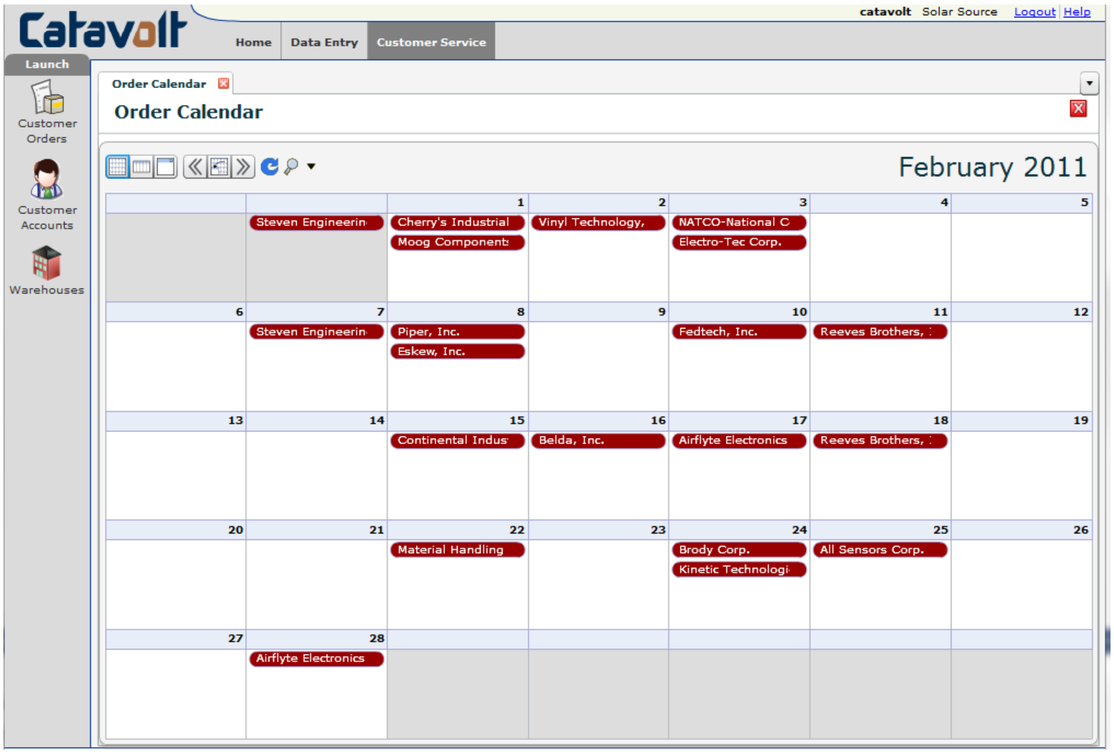


Figure 19: An example query displayed as a graph.

Or as a Calendar:



Order Calendar

February 2011

	Steven Engineerin	1 Cherry's Industrial Moog Component:	2 Vinyl Technology,	3 NATCO-National C Electro-Tec Corp.	4	5
6	7 Steven Engineerin	8 Piper, Inc. Eskew, Inc.	9	10 Fedtech, Inc.	11 Reeves Brothers, :	12
13	14	15 Continental Indus	16 Belda, Inc.	17 Airflyte Electronics	18 Reeves Brothers, :	19
20	21	22 Material Handling	23	24 Brody Corp. Kinetic Technologi	25 All Sensors Corp.	26
27	28 Airflyte Electronics					

Figure 20: An example query shown as a calendar.

Or as a Map

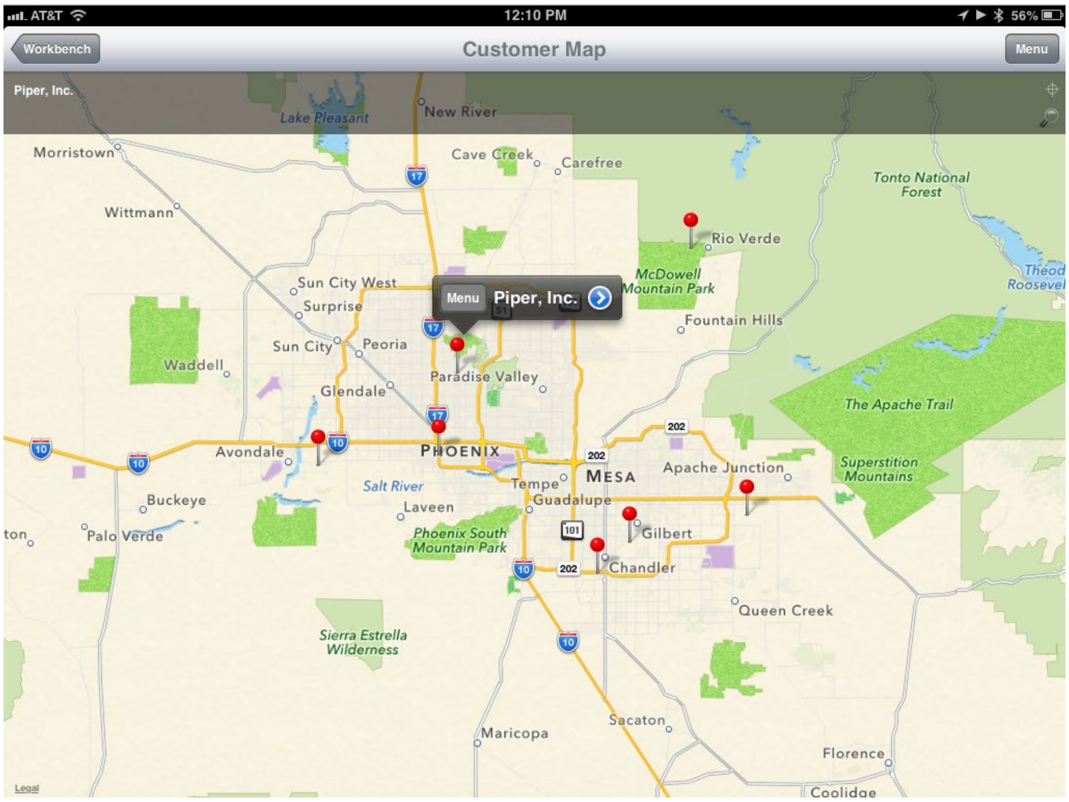


Figure 21: An example query shown as a calendar.

Or as a mobile Image Viewer

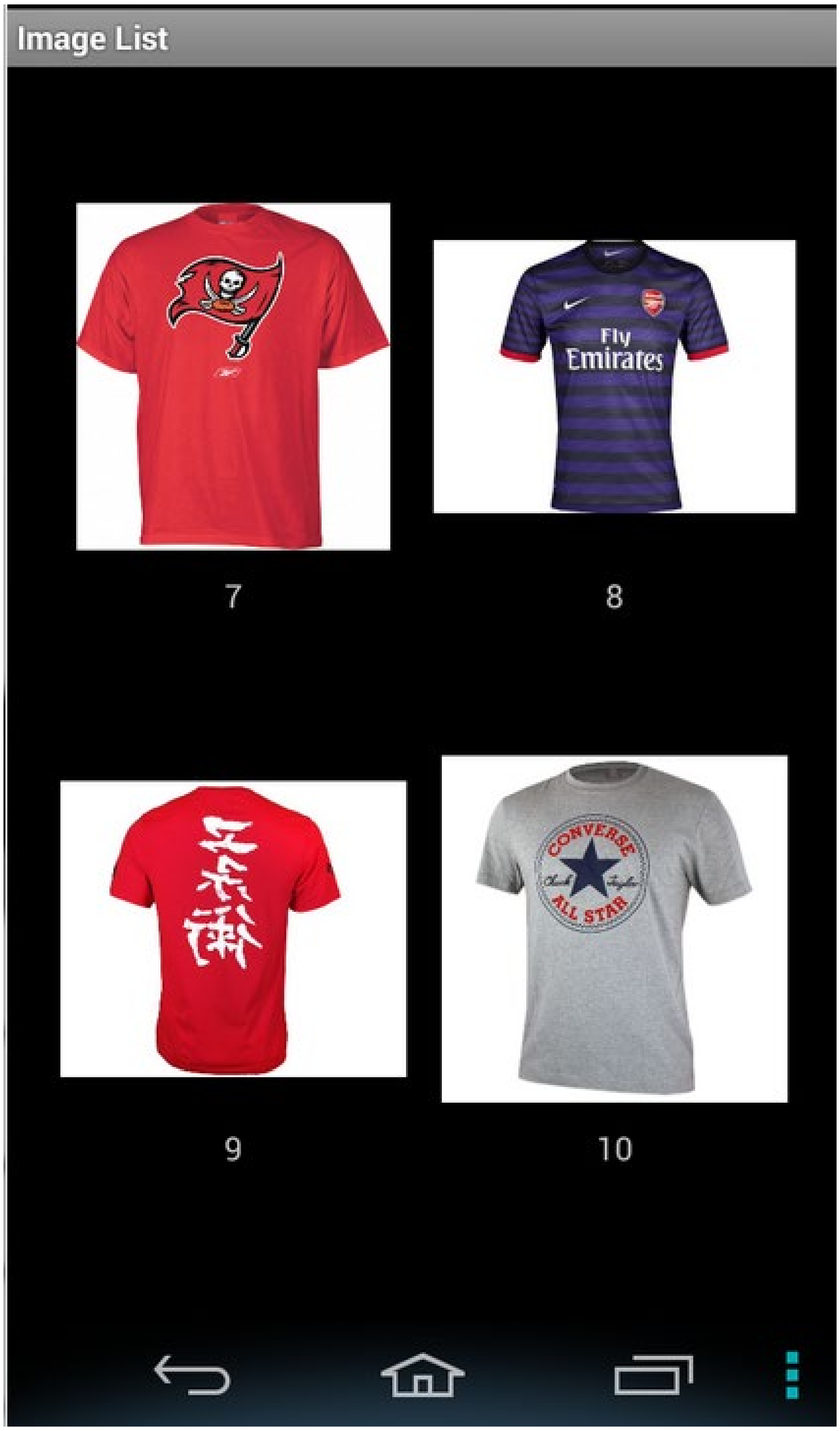
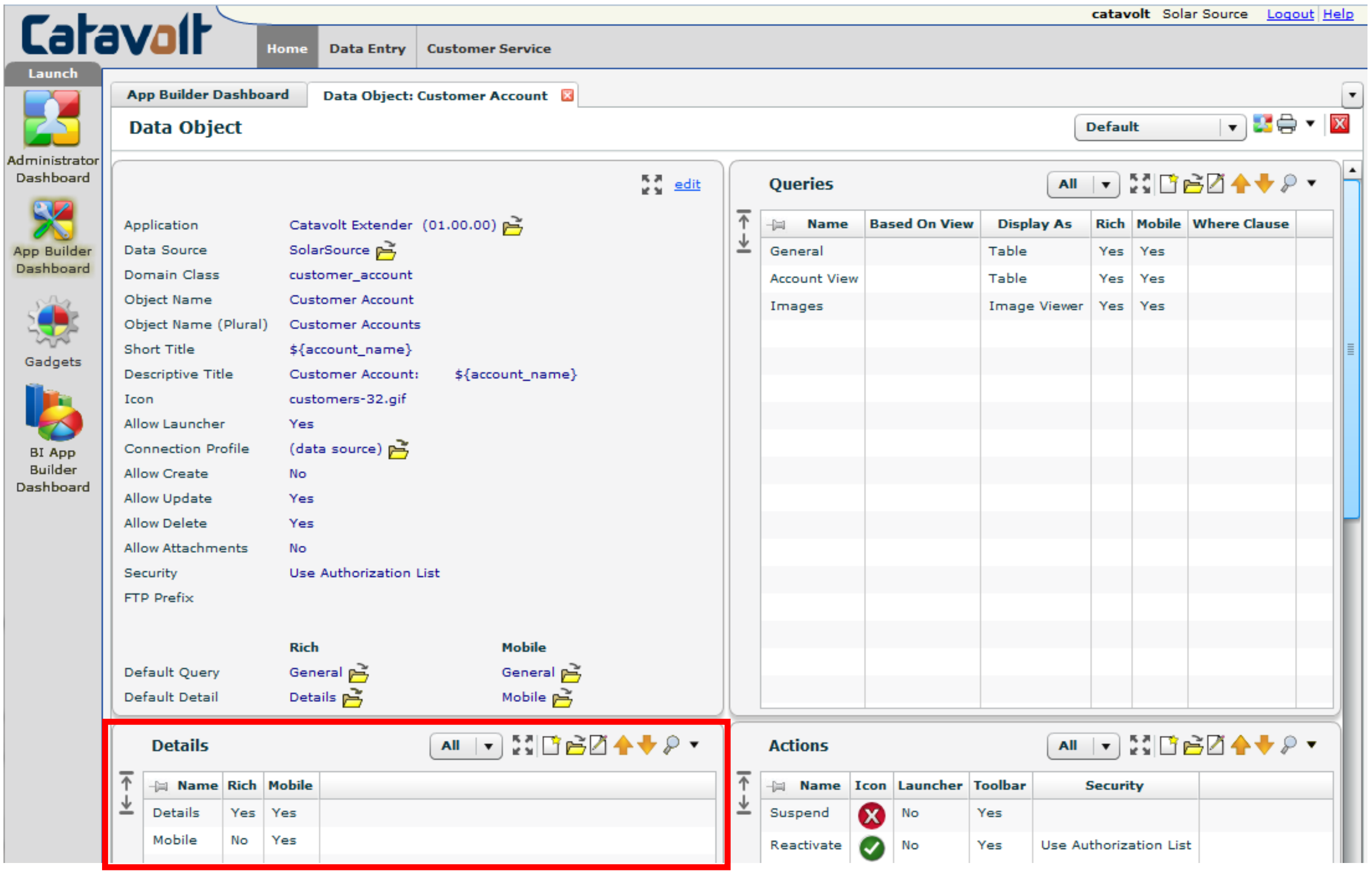


Figure 22: An example query shown as an image viewer.

Details

A Detail represents a single record and is usually accessed as a drill down from a query. A Detail allows you to more fully display the information about an individual record than what you can accomplish using a Query alone. As with Queries, you are allowed to define multiple Details for a particular Data Object. Each Detail can provide a different view over the Data Object so that you are not forced to attempt to put every possible field that may be of interest on a single Detail.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Short Title	${account_name}
Descriptive Title	Customer Account: ${account_name}
Icon	customers-32.gif
Allow Launcher	Yes
Connection Profile	(data source)
Allow Create	No
Allow Update	Yes
Allow Delete	Yes
Allow Attachments	No
Security	Use Authorization List
FTP Prefix	

	Rich	Mobile
Default Query	General	General
Default Detail	Details	Mobile

Queries

Name	Based On View	Display As	Rich	Mobile	Where Clause
General		Table	Yes	Yes	
Account View		Table	Yes	Yes	
Images		Image Viewer	Yes	Yes	

Details

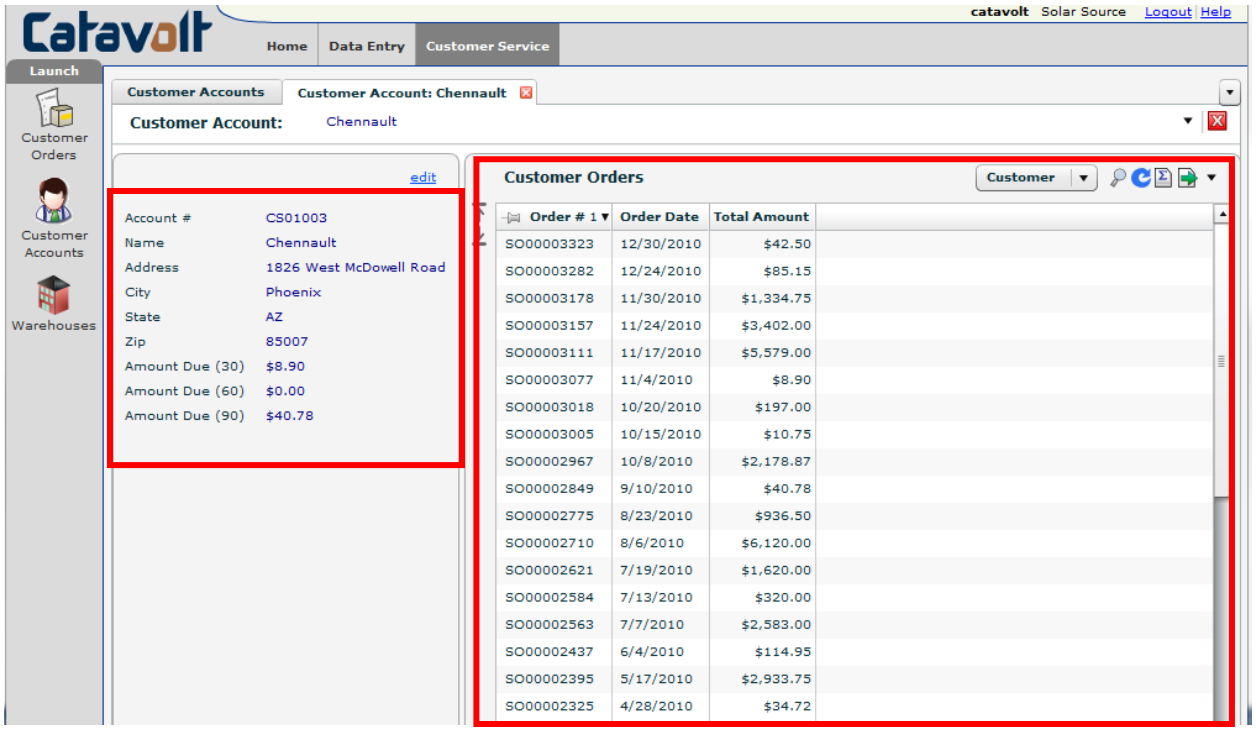
Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

Actions

Name	Icon	Launcher	Toolbar	Security
Suspend	✗	No	Yes	
Reactivate	✓	No	Yes	Use Authorization List

Figure 23: The Data Object "Default' detail with the Details query section highlighted.

Details contain one or more sections. Sections can contain properties from the record being displayed or queries of related records on other Data Objects.



Customer Account: Chennault

Account #	CS01003
Name	Chennault
Address	1826 West McDowell Road
City	Phoenix
State	AZ
Zip	85007
Amount Due (30)	$8.90
Amount Due (60)	$0.00
Amount Due (90)	$40.78

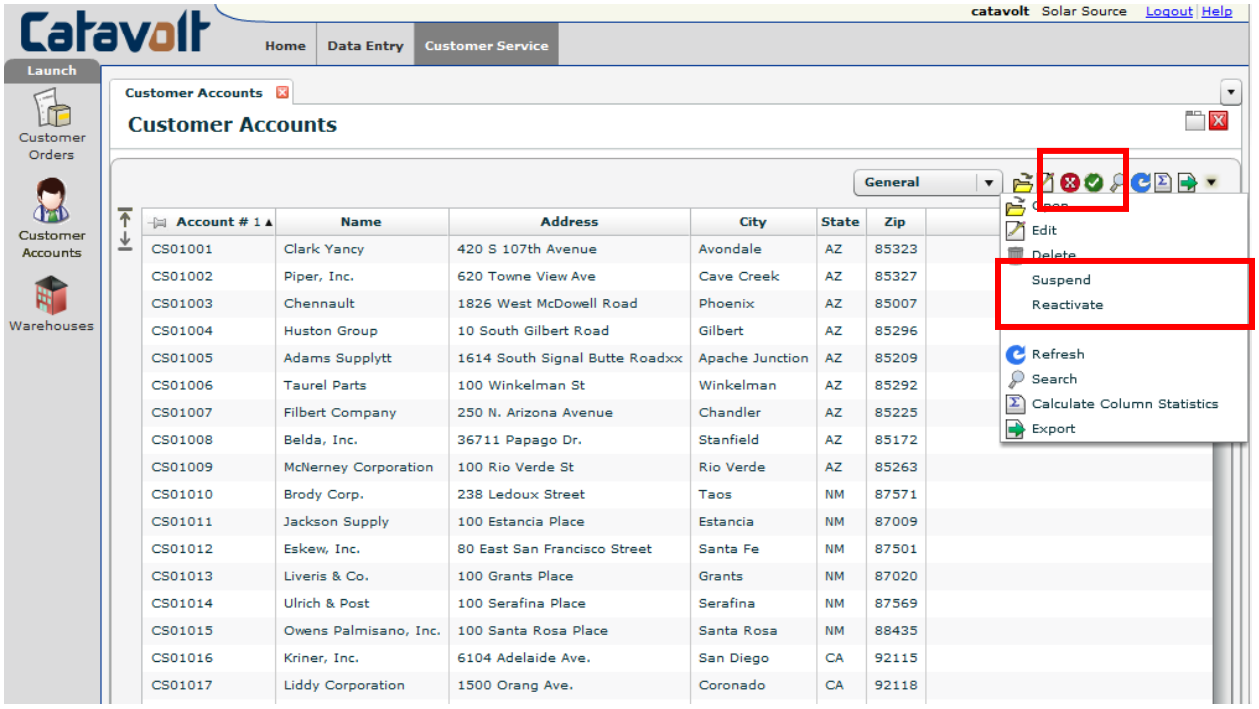
Customer Orders

Order # 1	Order Date	Total Amount
SO00003323	12/30/2010	$42.50
SO00003282	12/24/2010	$85.15
SO00003178	11/30/2010	$1,334.75
SO00003157	11/24/2010	$3,402.00
SO00003111	11/17/2010	$5,579.00
SO00003077	11/4/2010	$8.90
SO00003018	10/20/2010	$197.00
SO00003005	10/15/2010	$10.75
SO00002967	10/8/2010	$2,178.87
SO00002849	9/10/2010	$40.78
SO00002775	8/23/2010	$936.50
SO00002710	8/6/2010	$6,120.00
SO00002621	7/19/2010	$1,620.00
SO00002584	7/13/2010	$320.00
SO00002563	7/7/2010	$2,583.00
SO00002437	6/4/2010	$114.95
SO00002395	5/17/2010	$2,933.75
SO00002325	4/28/2010	$34.72

Figure 24: An example Details showing two sections. One section is showing local properties and another showing a list of related records.

Actions

An action represents a menu and/or toolbar action for a Data Object. Each action represents a unit of work that can be performed against a Data Object.

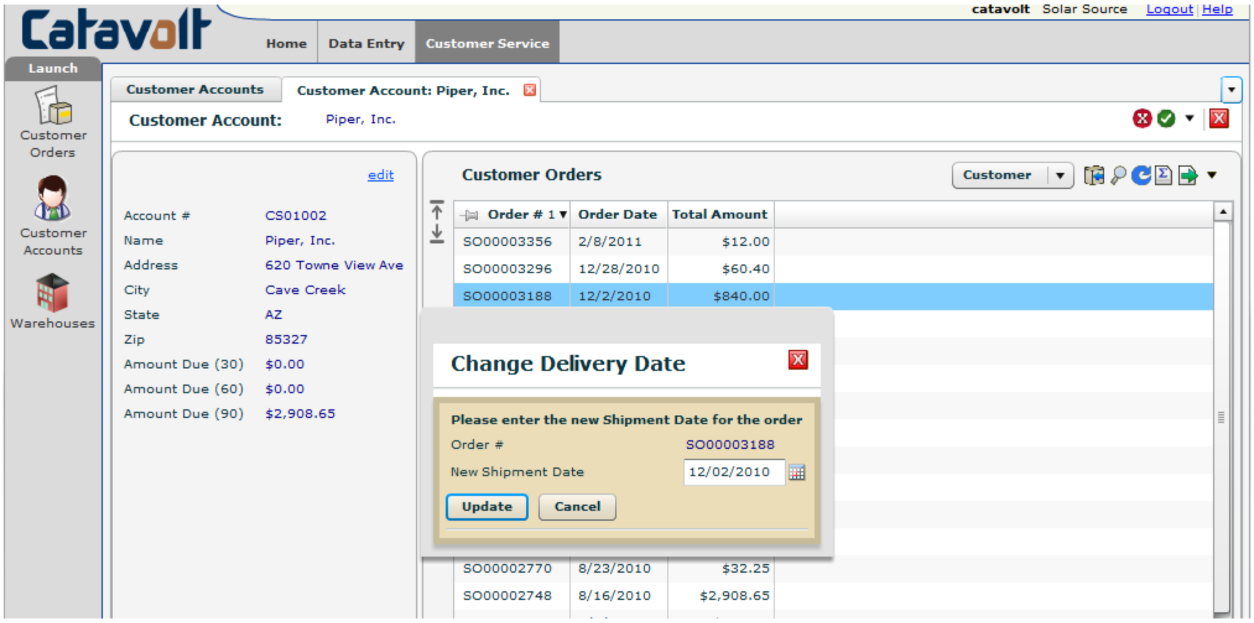


Customer Accounts

Account # 1 ▲	Name	Address	City	State	Zip
CS01001	Clark Yancy	420 S 107th Avenue	Avondale	AZ	85323
CS01002	Piper, Inc.	620 Towne View Ave	Cave Creek	AZ	85327
CS01003	Chennault	1826 West McDowell Road	Phoenix	AZ	85007
CS01004	Huston Group	10 South Gilbert Road	Gilbert	AZ	85296
CS01005	Adams Supplytt	1614 South Signal Butte Roadxx	Apache Junction	AZ	85209
CS01006	Taurel Parts	100 Winkelman St	Winkelman	AZ	85292
CS01007	Filbert Company	250 N. Arizona Avenue	Chandler	AZ	85225
CS01008	Belda, Inc.	36711 Papago Dr.	Stanfield	AZ	85172
CS01009	McNerney Corporation	100 Rio Verde St	Rio Verde	AZ	85263
CS01010	Brody Corp.	238 Ledoux Street	Taos	NM	87571
CS01011	Jackson Supply	100 Estancia Place	Estancia	NM	87009
CS01012	Eskew, Inc.	80 East San Francisco Street	Santa Fe	NM	87501
CS01013	Liveris & Co.	100 Grants Place	Grants	NM	87020
CS01014	Ulrich & Post	100 Serafina Place	Serafina	NM	87569
CS01015	Owens Palmisano, Inc.	100 Santa Rosa Place	Santa Rosa	NM	88435
CS01016	Kriner, Inc.	6104 Adelaide Ave.	San Diego	CA	92115
CS01017	Liddy Corporation	1500 Orang Ave.	Coronado	CA	92118

Figure 25: An example Data Object query showing actions on the toolbar and context menu.

You have the ability to prompt for data when running an Action. This data can include constant values as well as properties from the selected record:



Customer Account: Piper, Inc.

Account #	CS01002
Name	Piper, Inc.
Address	620 Towne View Ave
City	Cave Creek
State	AZ
Zip	85327
Amount Due (30)	$0.00
Amount Due (60)	$0.00
Amount Due (90)	$2,908.65

Customer Orders

Order # 1 ▼	Order Date	Total Amount
SO00003356	2/8/2011	$12.00
SO00003296	12/28/2010	$60.40
SO00003188	12/2/2010	$840.00
SO00002770	8/23/2010	$32.25
SO00002748	8/16/2010	$2,908.65

Change Delivery Date

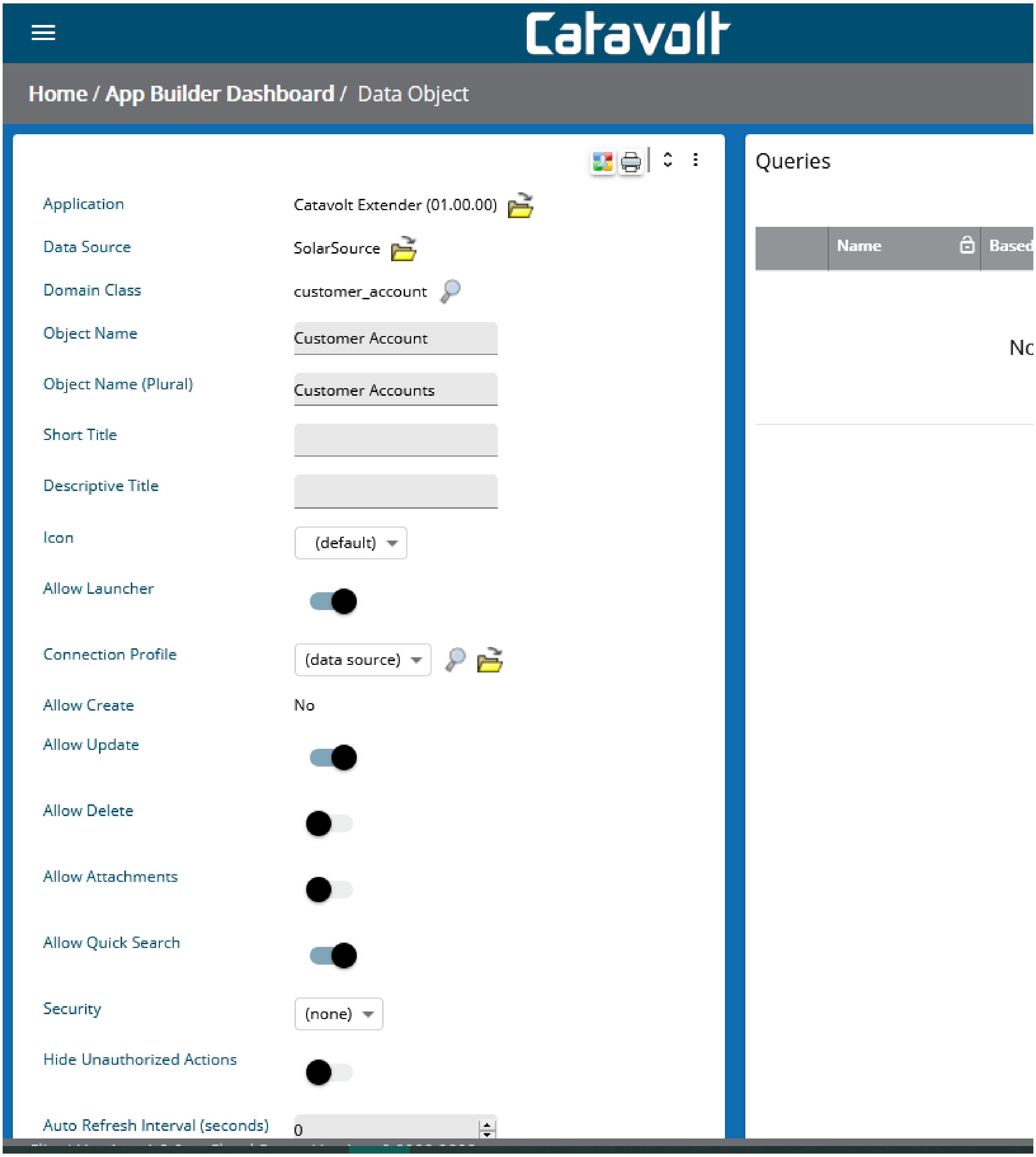
Please enter the new Shipment Date for the order

Order #	SO00003188
New Shipment Date	12/02/2010

Figure 26: An example action prompt

Actions can run a combination of API calls, SQL commands, stored procedure calls, native programs, etc., against the back-end system.

Creating a Data Object



Home / App Builder Dashboard / Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Short Title	
Descriptive Title	
Icon	(default)
Allow Launcher	☑
Connection Profile	(data source)
Allow Create	No
Allow Update	☑
Allow Delete	□
Allow Attachments	□
Allow Quick Search	☑
Security	(none)
Hide Unauthorized Actions	□
Auto Refresh Interval (seconds)	0

Queries

Name	Based

No

Figure 27: The Data Object create screen

When creating a Data Object, you must first select which Data Source this Data Object is for. This tells Xalt where to go to get the list of available Domain Classes to choose from.

You must then select a Domain Class. The Domain Class identifies the object or database table you wish to create a Data Object over.

Object Name is the singular name by which you want to identify this Data Object to users. Xalt may default this value at the time you pick a Domain Class if an Object Name is available.

Object Name (Plural) is the plural name by which you want to identify this Data Object to users. Xalt may default this value at the time you pick a Domain Class if an Object Name is available.

Short Title is used as a default identifier for this Data Object for options such as Graphs, Calendars, Object Lookups, etc. Xalt will default this value to the key fields separated by commas if you do not specify a value. See Appendix A: Specifying Messages and Substitution Values for more information about how to create a Short Title.

Descriptive Title is used as a heading identifier when displaying Details for this Data Object. Xalt will default this value to Object Name followed by the key fields if you do not specify a value. See Appendix A: Specifying Messages and Substitution Values for more information about how to create a Descriptive Title.

Icon allows you to specify the default image to display when adding this Data Object to a Workbench. You can choose the default Hexagon image or any other image you have uploaded. See Chapter 2: Data Sources for more information on how to upload images.

Allow Launcher allows you to specify whether or not this Data Object is available to be added to a Workbench. If you select this checkbox the Data Object will appear in the Top Level query when viewing a list of Data Objects. A query launcher will also be available to add to workbenches. The query launcher will only be available once one or more queries have been created.

Connection Profile allows you to specify how Xalt should connect to the back-end system when querying and maintaining this Data Object. The available options are:

- (data source) –
  Use the Primary User ID and Primary Password for the Data Source to connect to the back-end system.
- (current user) –
  Use the Hexagon User ID and Password that the user supplies when logging into Xalt to connect to the back-end system. This is also known as the “pass-thru” option, as the User ID / Password is passed through Xalt into the back-end system
- specific value –
  Use a defined Connection Profile for the Data Source to connect to the back-end system.

Allow Create allows you to specify whether instances of this Data Source support the standard Create action. Selecting this option will automatically add a create action to the toolbar and context menu if a maintainable detail section exists. Note that this option is currently only available for Infor XA System-Link, SAP, and Salesforce Data Sources.

Allow Update allows you to specify whether instances of this Data Source support the standard Update action. Selecting this option will automatically add an update action to the toolbar and context menu if a maintainable detail section exists.

Allow Delete allows you to specify whether instances of this Data Source support the standard Delete action. Selecting this option will automatically add a delete action to the toolbar and context menu.

Allow Attachments allows you to view and create attachments for your Data Object. This option is available for Data Objects running Data Sources that have enabled FTP. See Appendix D: Attachments for more information about using Attachments.

Allow Quick Search allows you use a keyword (Google-like) search for this Data Object. Extender will submit a case-insensitive contains query on the columns in the view. If the search contains multiple words, these will be OR'ed together in the query. You may decide to turn off Quick Search for large datasets where this kind of search would be resource intensive. The XHA Advanced Search menu option will still be available even if Quick Search is not allowed. Note that the Search menu option for mobile clients uses the same search as Quick Search, so turning Quick Search off on the Data Object will also remove the Search menu option for mobile clients.

Security allows you to secure this Data Object to be accessed by specific Security Roles. The available options are:

- (none) – Do not add extra security this Data Object. Anyone who can launch or navigate to this Data Object will have access to it.
- Use authorization list – This option allows you to define a Permission list that contains one or more Security Roles and the actions that each is allowed to perform

See Chapter 10: Security for more information about object security in Xalt.

Hide Unauthorized Actions specifies how to handle Actions that the current user is not authorized to perform. If this option is selected, these actions will not appear on the user's menu or toolbar. If this option is not selected, these actions will appear on the menu and toolbar but will return an error if the user attempts to run the action.

See Chapter 10: Security for more information about action security in Xalt.

Auto Refresh Interval (seconds) s pecifies the amount of time that should pass before the window automatically refreshes itself. Allowable values are 0 (no auto-refresh) or between 5 and 14400 seconds (4 hours). The default value is 0 (no auto-refresh). Note that all Data Objects have a menu option to manually refresh the window regardless of the refresh interval setting. Turning on auto-refresh has the potential to greatly increase the load on your back-end server depending on the time set and the number of users accessing this Data Object.

Apply Auto Refresh Interval To specifies whether the auto refresh timer applies to just Query windows, Detail windows, or both. The default value is Queries Only. When Auto Refresh is applied to a Detail, the setting is used for the entire Detail window, including all sub-sections. If the Detail contains Query Sections that point to other Data Objects that also have auto-refresh enabled, the refresh timers for those Data Objects are ignored. The refresh timers for those Data Object will be applied again when drilling into Details for those Data Objects.

FTP Prefix allows you to specify an extra path in addition to the FTP Path of your Data Source. If specified, this extra path will be appended to the FTP Path for attachments on this Data Object and is used to allow attachments for different Data Objects to exist in separate folders. The path should be in FTP syntax with / as folder separators.

Offline Access allows you to specify whether this Data Object can be shown while offline. The available options are:

Explicit – Allow the end user to choose which records are taken offline

Implicit – Take all records offline

Online Only – Do not allow records to be taken offline

Please see the **Xalt Mobility – Offline Guide** for more information about Offline Access.

GML/Form Alias allows you to specify an alias when using this Data Object with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

External Resource Prefix allows you to specify common prefix to allow images/signatures/attachments managed by Hexagon to be shared across multiple Data Objects. In general, images/signatures/attachments are associated with an individual Data Object via a unique token. If you copy the Data Object or have multiple Data Objects over the same set of data, they will not share the images/signatures/attachments. Entering a common value for External

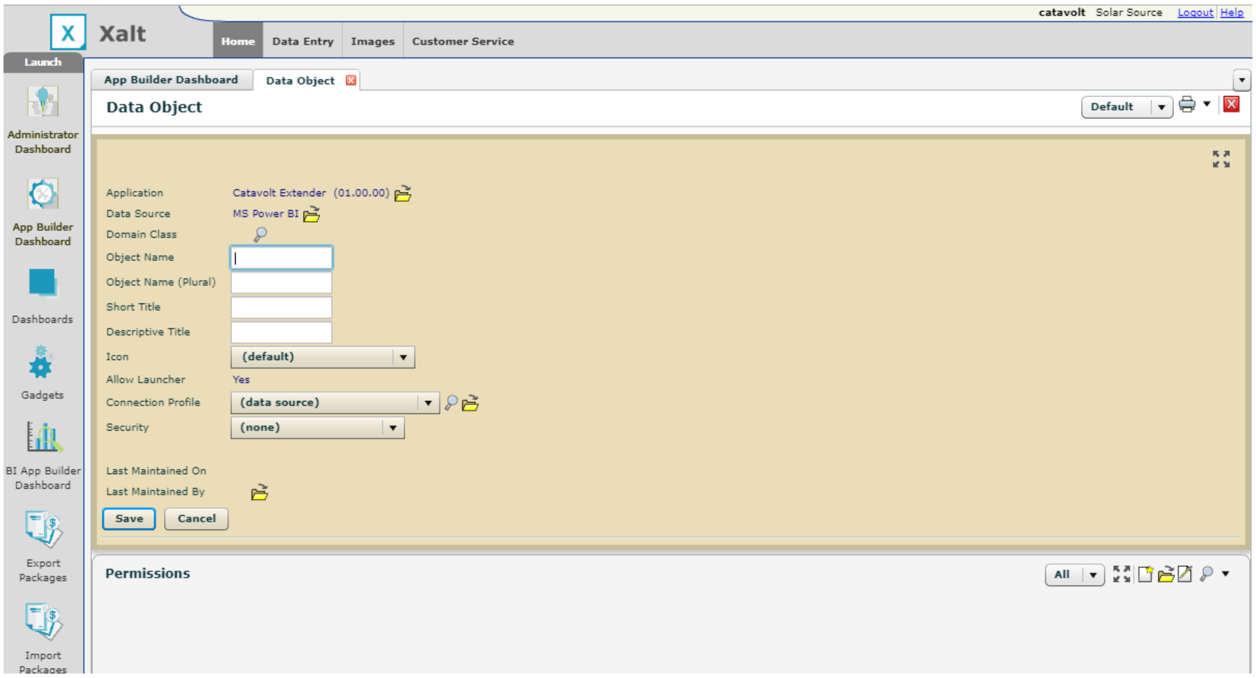
Resource Prefix for these Data Objects will start storing images/signatures/attachments in common. Your Hexagon sales representative has more information about using External Resource Prefixes.

Default Rich/Mobile Query allows you specify which Query will be initially displayed when you bring up a list of this Data Object. Note that Rich (Web and Gadgets) and Mobile can each have their own default Query.

Default Rich/Mobile Detail allows you specify which Detail will be initially displayed when you bring up an instance of this Data Object. Note that Rich (Web and Gadgets) and Mobile can each have their own default Detail.

Since Xalt has a minimal integration with Power BI, the user will see a much smaller set of options when viewing Data Objects for Power BI Data Sources. The following are differences between Power BI Data Objects and Data Objects for other Data Sources:

1) The Data Object is missing all sections on the Default view except the Permissions section.
2) The Object Structure view is missing (only Default and Used By are available).
3) Allow Launcher is hardcoded to Yes and cannot be changed.
4) A number of other properties and menu options do not appear.



Data Object

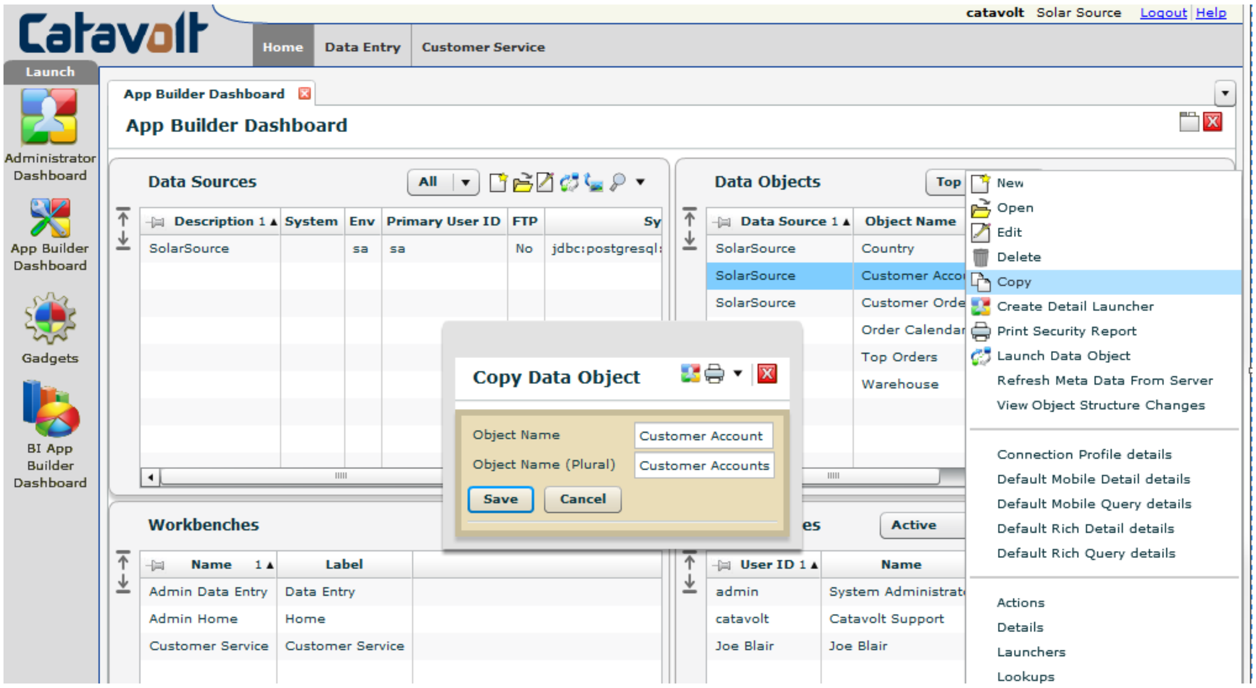
Application	Catavolt Extender (01.00.00)
Data Source	MS Power BI
Domain Class	
Object Name	
Object Name (Plural)	
Short Title	
Descriptive Title	
Icon	(default)
Allow Launcher	Yes
Connection Profile	(data source)
Security	(none)
Last Maintained On	
Last Maintained By	

Permissions

Figure 28: The Data Object create screen for Power BI Data Objects

Copying a Data Object

You may have instances where you need to make a copy of an existing Data Object to deploy to a different set of users. You can select the Copy menu option to accomplish this. When Copying a Data Object, you will be prompted to supply the new Object Name and Object Name (Plural). An exact copy of this Data Object along with all of its components will be made.



Description 1 ▲	System	Env	Primary User ID	FTP	Sy
SolarSource		sa	sa	No	jdbc:postgresql:

Data Source 1 ▲	Object Name
SolarSource	Country
SolarSource	Customer Acco
SolarSource	Customer Orde
	Order Calendar
	Top Orders
	Warehouse

Copy Data Object

Object Name: Customer Account

Object Name (Plural): Customer Accounts

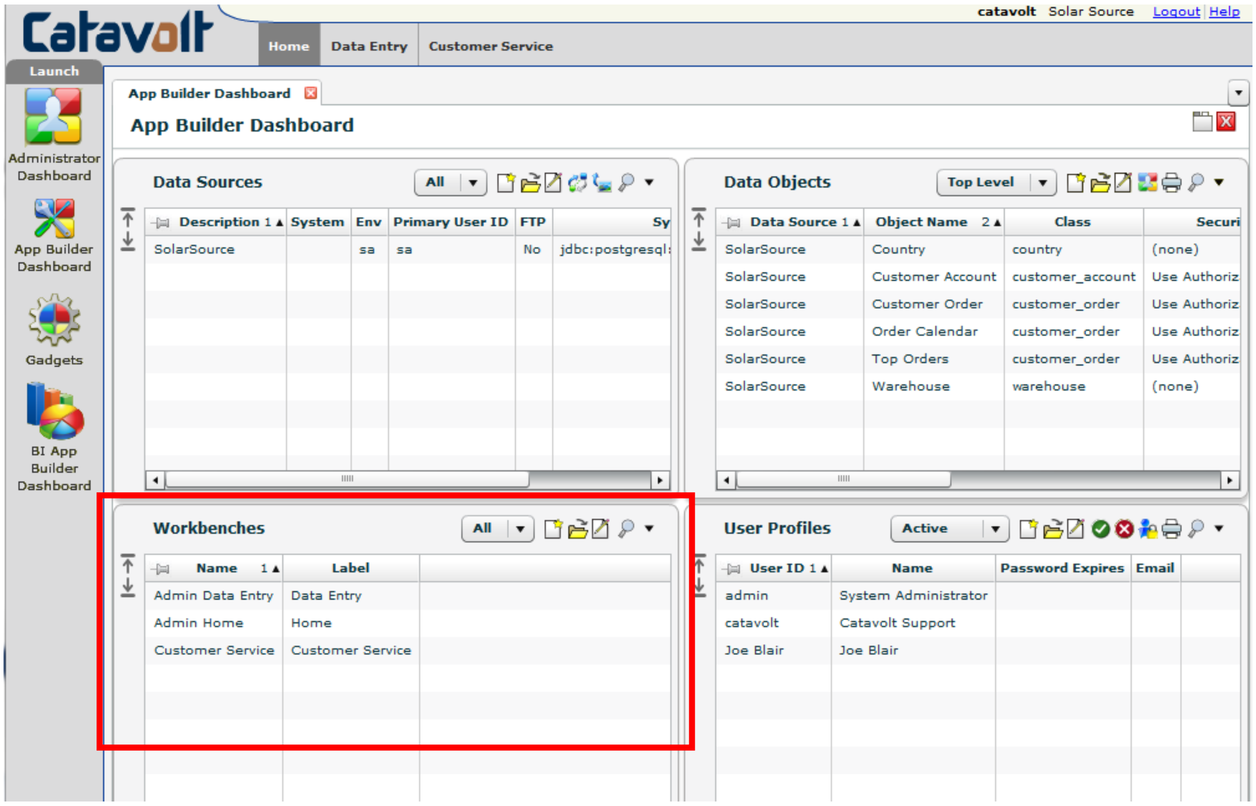
Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

User ID 1 ▲	Name
admin	System Administrat
catavolt	Catavolt Support
Joe Blair	Joe Blair

Figure 29: The Copy Data Object prompt.

Deploying a Data Object

You must deploy the data object for users to gain access to it. To accomplish this, go to the list of Workbenches inside the App Builder Dashboard.



Description 1 ▲	System	Env	Primary User ID	FTP	Sy
SolarSource		sa	sa	No	jdbc:postgresql:

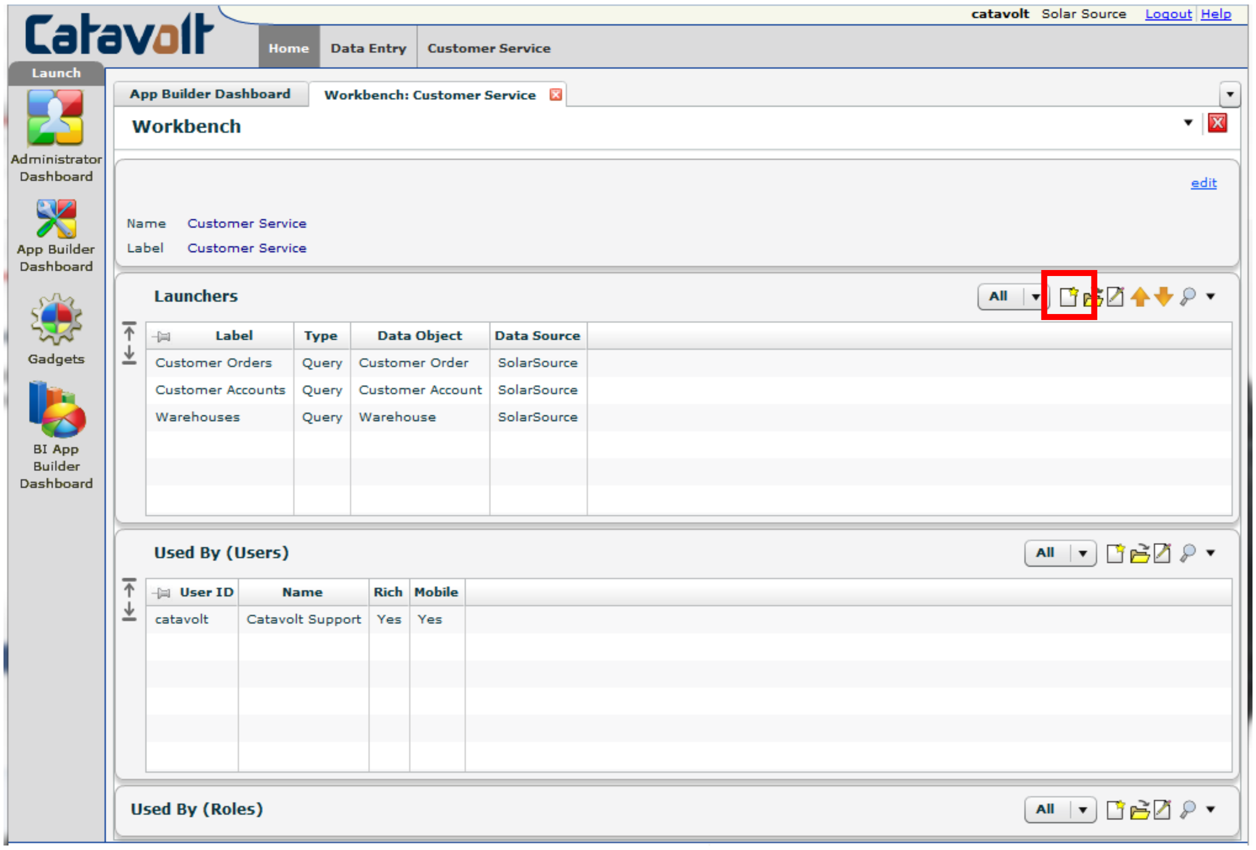
Data Source 1 ▲	Object Name 2 ▲	Class	Securi
SolarSource	Country	country	(none)
SolarSource	Customer Account	customer_account	Use Authoriz
SolarSource	Customer Order	customer_order	Use Authoriz
SolarSource	Order Calendar	customer_order	Use Authoriz
SolarSource	Top Orders	customer_order	Use Authoriz
SolarSource	Warehouse	warehouse	(none)

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Figure 30: The App Builder Dashboard with the Workbenches query section highlighted.

You may choose to create a new workbench or add the Data Object to an existing workbench. Go into the Details for the Workbench and press the Create button on the Launchers list



Workbench

Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource

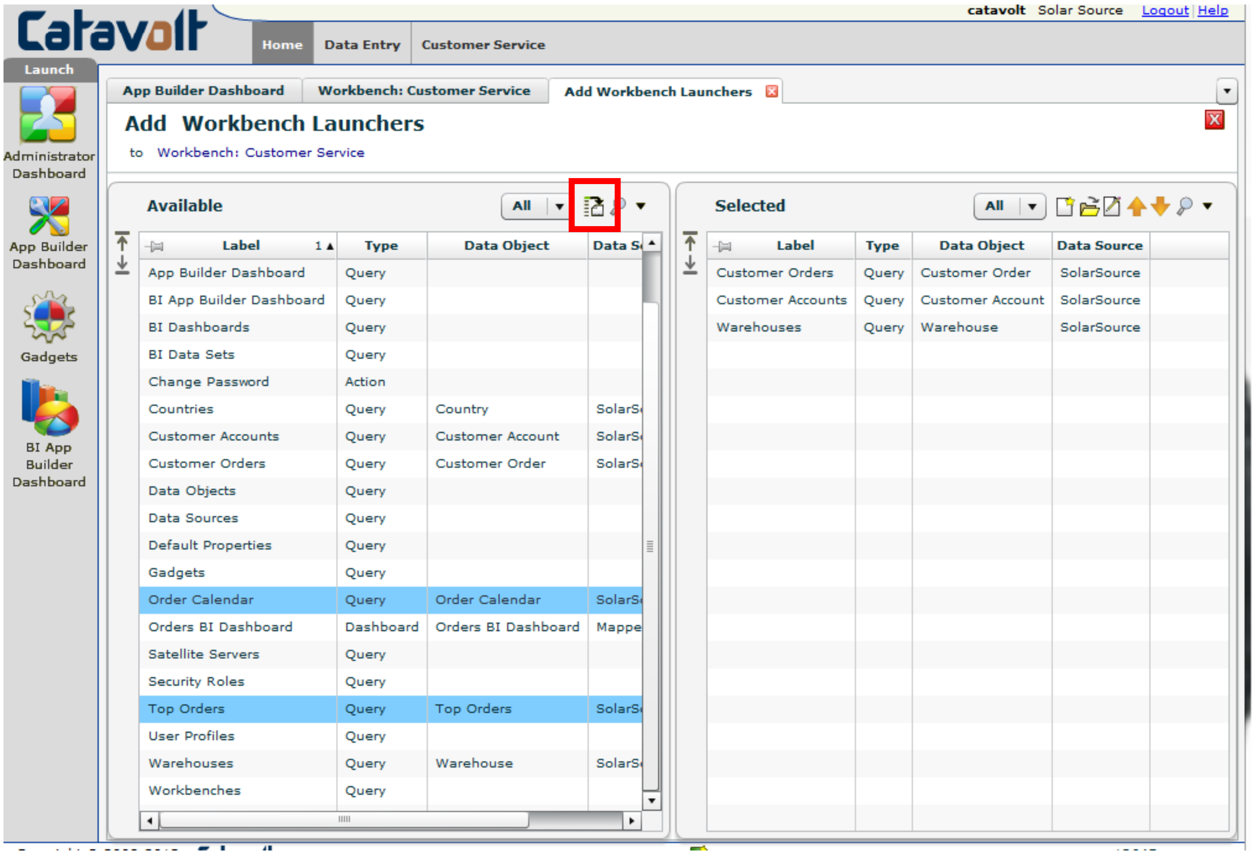
Used By (Users)

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes

Used By (Roles)

Figure 31: Workbench details

Select the Launcher(s) you want to add to the Workbench and choose the Add menu option



Add Workbench Launchers

to Workbench: Customer Service

Available

Label	Type	Data Object	Data S
App Builder Dashboard	Query		
BI App Builder Dashboard	Query		
BI Dashboards	Query		
BI Data Sets	Query		
Change Password	Action		
Countries	Query	Country	SolarS
Customer Accounts	Query	Customer Account	SolarS
Customer Orders	Query	Customer Order	SolarS
Data Objects	Query		
Data Sources	Query		
Default Properties	Query		
Gadgets	Query		
Order Calendar	Query	Order Calendar	SolarS
Orders BI Dashboard	Dashboard	Orders BI Dashboard	Mappe
Satellite Servers	Query		
Security Roles	Query		
Top Orders	Query	Top Orders	SolarS
User Profiles	Query		
Warehouses	Query	Warehouse	SolarS
Workbenches	Query		

Selected

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource

Figure 32: The add workbench launcher view showing the Add menu action

Note that in order for your Data Object to appear in the Available list, the Allow Launcher checkbox must be selected for the Data Object. In addition, the Data Object must have at least one Query defined for it as well.

Once you have added a Data Object to a Workbench, the icon will appear to the end user the next time they log into Xalt:

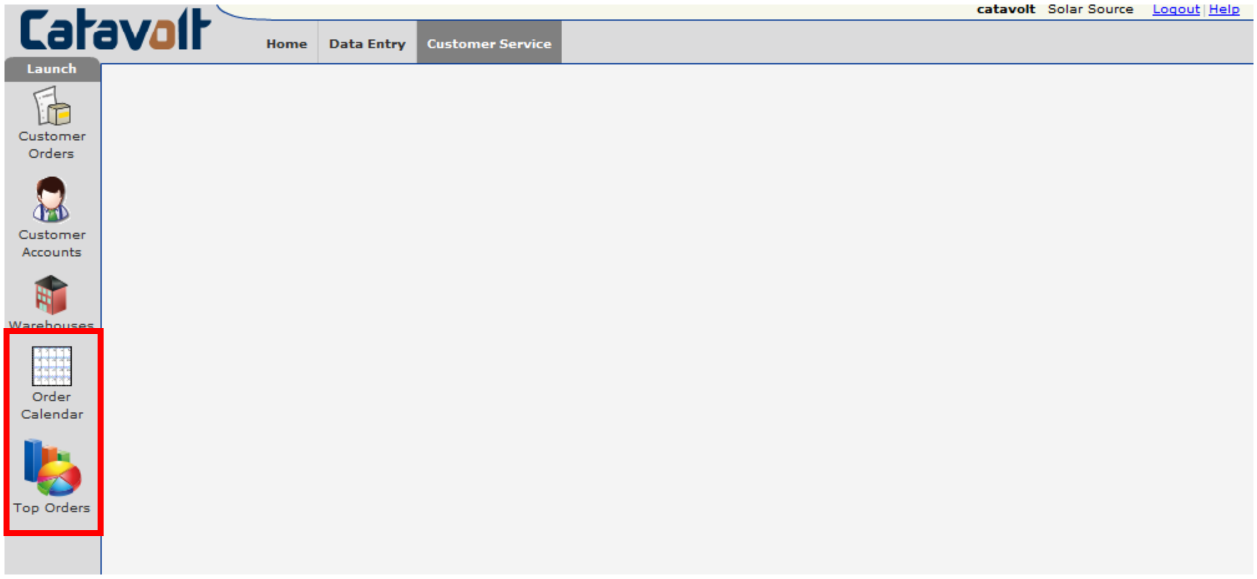
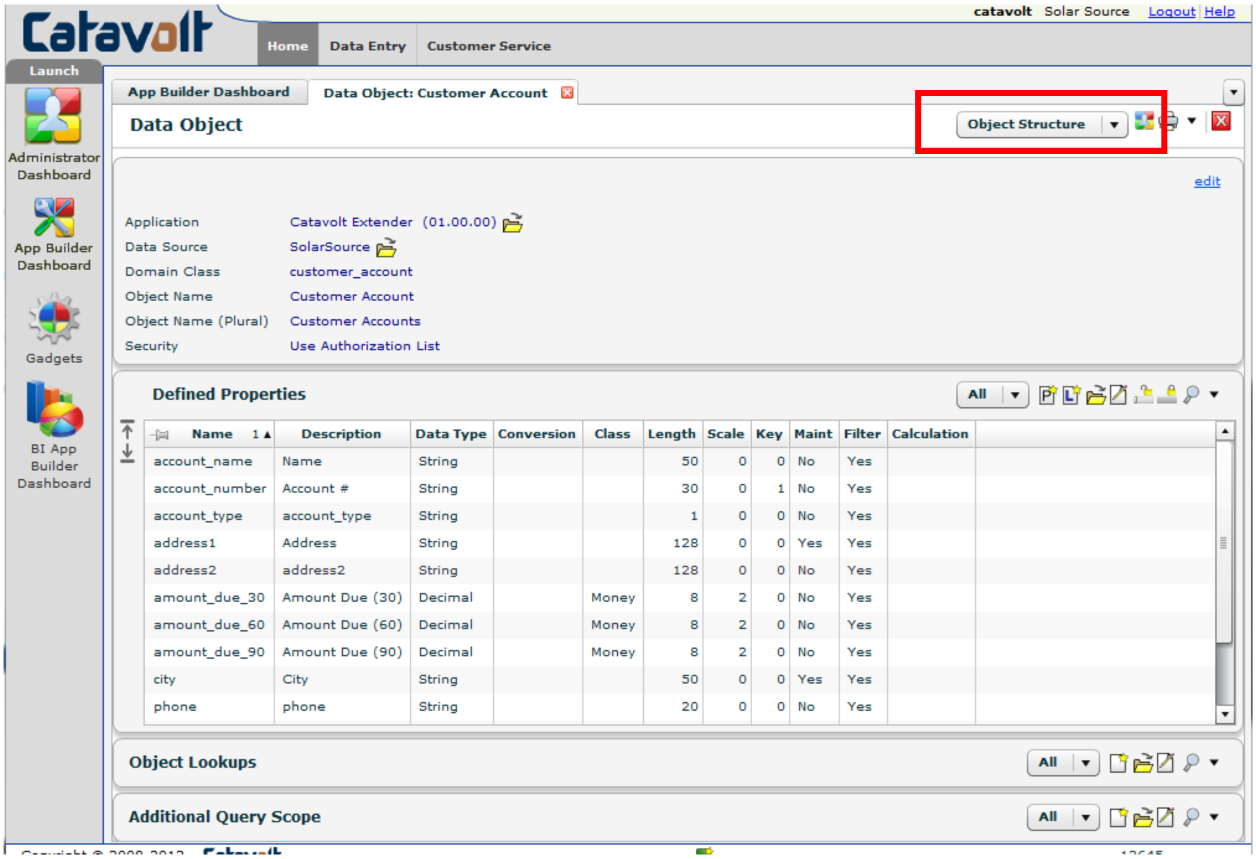


Figure 33: Xalt runtime as seen by the end user showing the Launchers that were added to the Workbench.

Data Object Defined Properties

Data Object properties are used on queries, details, actions and substitution messages. They typically represent columns in a database table. The Defined Properties section on the Object Structure detail of a Data Object shows a list of all properties that have been defined for this Data Object. As you add properties to Queries, Details, etc., they are automatically added or to or updated on the Defined Properties list.



Data Object

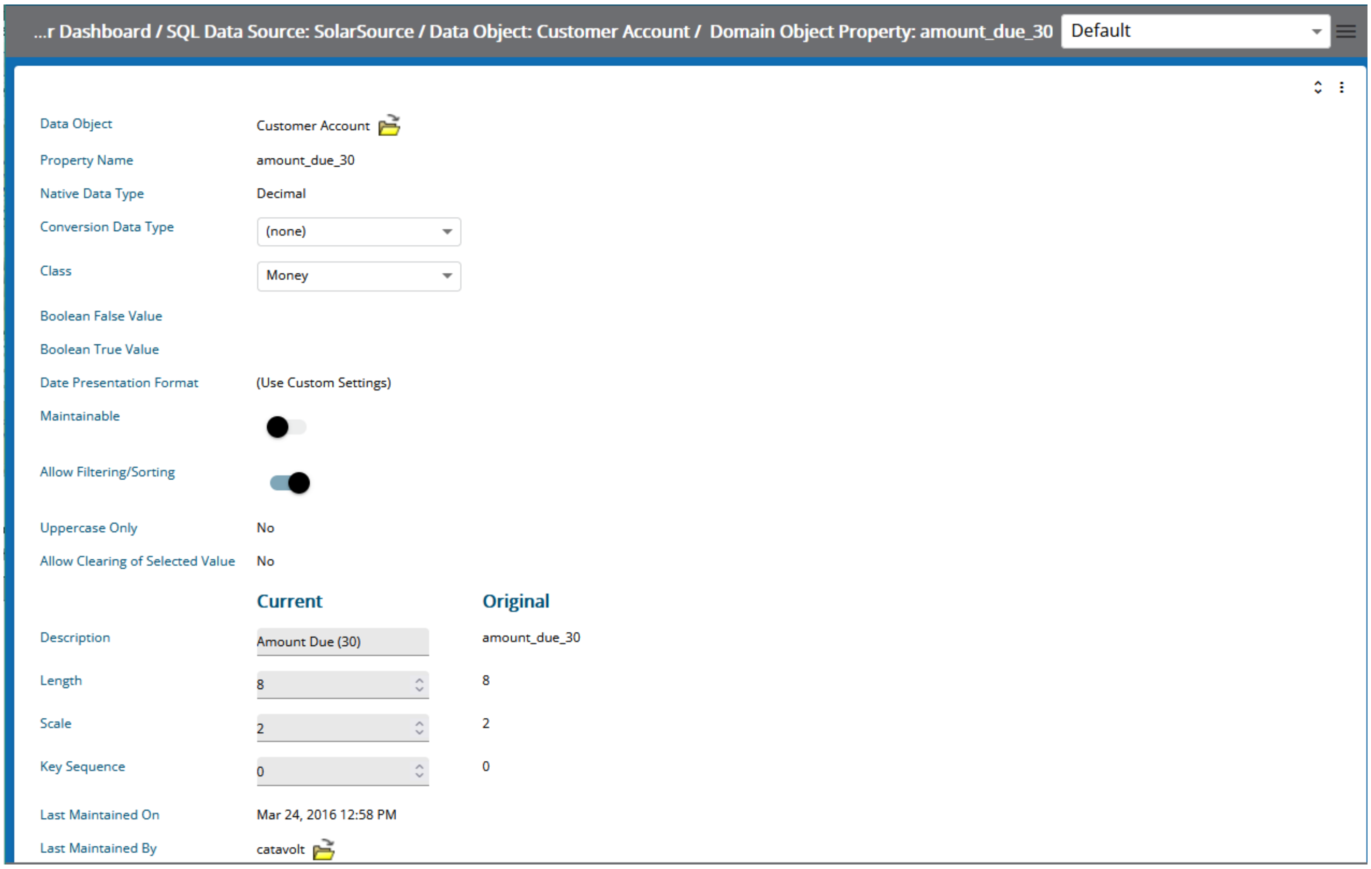
Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Security	Use Authorization List

Defined Properties

Name 1 ▲	Description	Data Type	Conversion	Class	Length	Scale	Key	Maint	Filter	Calculation
account_name	Name	String			50	0	0	No	Yes	
account_number	Account #	String			30	0	1	No	Yes	
account_type	account_type	String			1	0	0	No	Yes	
address1	Address	String			128	0	0	Yes	Yes	
address2	address2	String			128	0	0	No	Yes	
amount_due_30	Amount Due (30)	Decimal		Money	8	2	0	No	Yes	
amount_due_60	Amount Due (60)	Decimal		Money	8	2	0	No	Yes	
amount_due_90	Amount Due (90)	Decimal		Money	8	2	0	No	Yes	
city	City	String			50	0	0	Yes	Yes	
phone	phone	String			20	0	0	No	Yes	

Figure 34: The Object Structure detail of a Data Object

When you display the definition for a Property, there are 2 Details that you can select from. The Default Detail shows you all the information about the Property. In most cases, you do not need to go to the Defined Properties list to update a property. However, there are circumstances where you need to edit the definition of a property. Some examples of this would be to define key fields for tables that do not define a unique key, change the data type of a property, or add special behavior to the property.



...r Dashboard / SQL Data Source: SolarSource / Data Object: Customer Account / Domain Object Property: amount_due_30 | Default

Data Object	Customer Account	
Property Name	amount_due_30	
Native Data Type	Decimal	
Conversion Data Type	(none)	
Class	Money	
Boolean False Value		
Boolean True Value		
Date Presentation Format	(Use Custom Settings)	
Maintainable	Off	
Allow Filtering/Sorting	On	
Uppercase Only	No	
Allow Clearing of Selected Value	No	
	Current	**Original**
Description	Amount Due (30)	amount_due_30
Length	8	8
Scale	2	2
Key Sequence	0	0
Last Maintained On	Mar 24, 2016 12:58 PM	
Last Maintained By	catavolt	

Figure 35: Editing a Domain Object Property

When editing the definition of a Property, you have the following options:

Data Object specifies which Data Object this Property is defined on. This value cannot be changed.

Property Name specifies the name this Property is known by. It typically corresponds to the column name in the database table. This value cannot be changed.

Native Data Type is the property type as known to the back-end system. This value cannot be changed.

Conversion Data Type is the property type as presented to the Xalt user. Note that this type can be different from the Native Data Type.

Decimal or Long types can have the following Conversion Data Types:

- Boolean – Presented to the Xalt User as a true/false switch. Stored in the back-end as a number specified by the Boolean True Value and Boolean False Value properties.
- Date (CYYMMDD) – Presented to the Xalt User as a Date field. Stored in the back-end system as a 7-digit value (C=Century (20th=0, 21st=1,etc), YY=2 digit Year, MM=Month, DD=Day)
- Date (YYYYMMDD) – Presented to the Xalt User as a Date field. Stored in the back-end system as an 8-digit value (YYYY=4 digit Year, MM=Month, DD=Day)

- Time (HHMMSS) – Presented to the Xalt User as a Time field. Stored in the back-end system as a 6-digit value (HH=Hour (24--hour format), MM=Minutes, SS=Seconds)
- Time (HHMM) – Presented to the Xalt User as a Time field. Stored in the back-end system as a 4-digit value (HH=Hour (24--hour format), MM=Minutes)

String types can have the following Conversion Data Types:

- Boolean – Presented to the Xalt User as a true/false switch. Stored in the back-end as a String specified by the Boolean True Value and Boolean False Value properties.

Timestamp types can have the following Conversion Data Types:

- Date – Presented to the Xalt User as a Date field. Stored in the back--end system as a Timestamp field with the Time portion set to 00:00:00.
- Time – Presented to the Xalt User as a Time field. Stored in the back--end system as a Timestamp field with the Date portion set to 1/1/1970.

Note that Conversion Data Types for Timestamps are typically used for databases such as SQL Server where only a single datetime database type exists. Without separate data and time types, a way is needed to present the significant portion of the datetime type.

Class specifies special formatting options used to display values for this property.

Decimal or Long types can have the following Classes:

- Attachment Counter – Keeps a count of the number of attachments that are associated with a given record. This is available for non-Infor XA Data Objects that allow Attachments. See Appendix D: Attachments for more information about using Attachments.
- Barcode – Used to indicate a Barcode value
- Barcode (Auto-Scan) – Prevents a mobile device's soft keyboard from appearing on screen when this field gets normal focus. Manually selecting the field will subsequently bring up the soft keyboard if manual entry is required.
- Money – Displayed as a monetary value with a $.
- NFC – Used to indicate a value from an NFC tag
- Percent – Displayed as a percent value with percent sign. The value should be stored in the database in place value (e.g. .063). Xalt will automatically multiply and render the value as a percent (e.g. 6.3%)
- Unformatted Number – Displayed without thousands separators. This is typically used for non-quantity fields that are stored in the database as numeric fields (such as account numbers).
- Latitude – Used to indicate the latitude value for a location (used in Mapping).
- Longitude – Used to indicate the longitude value for a location (used in Mapping).

String types can have the following Classes:

- Barcode – Used to indicate a Barcode value
- E-mail – Used to indicate an e-mail address.
- NFC – Used to indicate a value from an NFC tag
- URL – Used to indicate a web URL.
- Password – Used to indicate a password. Password fields are display using ***** chars.
- Latitude/Longitude – Used to indicate the latitude/longitude value for a location (used in Mapping).

- Telephone – Used to indicate a Phone Number.
- Text – Used to indicate the String value should be rendered using a multi-line text control instead of a single-line entry field on details and actions. Note that String fields with a length greater than 255 will continue to render using a multi-line text control as they did in V2.

Large Binary types can have the following Classes:

- Signature – Used to indicate a signature value to capture (as opposed to an image)

Boolean False Value – For String and Numeric values that have a Conversion Data Type of Boolean, this indicates the actual value that should be read from / written to the back-end system for false property values. The description 'No' will be shown to users. Note that this is not needed for native Boolean fields, as these fields already support native false values.

Boolean True Value - For String and Numeric values that have a Conversion Data Type of Boolean, this indicates the actual value that should be read from / written to the back-end system for true property values. The description 'Yes' will be shown to users. Note that this is not needed for native Boolean fields, as these fields already support native true values.

Date Presentation Format specifies special formatting options used to display Date values in read mode. This value is only available for Date and Timestamp properties.

- (Use Custom Settings) – Use the format specified in Custom Settings.
- Short – Force the client to use a Short Date format (actual format is dependent on client's locale settings)
- Long – Force the client to use a Long Date format (actual format is dependent on client's locale settings)

Maintainable specifies whether the value for this property can be changed by the end user. Note that if the back-end system has determined that this property can never be maintained, this value will be disabled.

Allow Filtering/Sorting specifies whether the end user can use this property to change the filter and/or sort they are using for the current query. This is typically used for back-end systems that may return logical properties that cannot participate in filtering and sorting.

Uppercase Only specifies whether Xalt should automatically convert any data entered by the end user to upper case when sending it to the back-end system. This is used for String properties where the back-end system requires the data to be upper case but does not convert entered data to upper case itself.

Allow Clearing of Selected Value specifies that an empty (null) value should be added as an option to the property's dropdown list to allow the user to "unselect" a value for this property. This field is disabled unless the property has a dropdown list (at least one Property Value exists, the property is the recipient of an Object Lookup with Combo Box, or the Property has a Native Data Type of Code Ref).

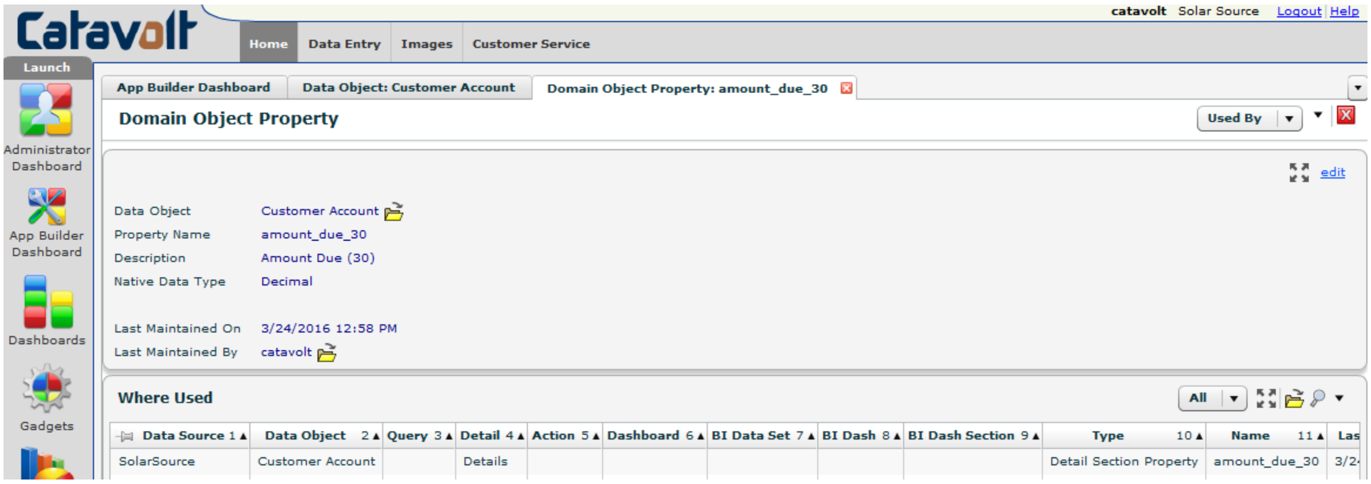
Description allows you to set a default description to be used when this property is added to a Query, Detail, Action, etc.

Length allows you to override the property's length. Xalt will attempt to determine the correct value from the back-end system when the field is first loaded. You can specify shorter lengths for numeric and string fields in cases where your field may be longer than the data you want your users to enter. Note that some data sources such as Infor XA System-Link allow you to set this value to be shorter or longer than the original value as some versions of System-Link do not return a property's actual length.

Scale allows you to override the number of decimals for a numeric property. Xalt will attempt to determine the correct value from the back-end system when the field is first loaded. You can specify shorter scales for numeric fields in cases where your field may have more decimals than the data you want your users to enter. Note that some data sources such as Infor XA System-Link allow you to set this value to be shorter or longer than the original value as some versions of System-Link do not return a property's actual scale.

Key Sequence allows you to set a primary key for a Data Object. Xalt requires that all Data Objects define a unique key. A unique key is a combination of fields that, when used together, guarantees that no two records have the same values for the fields. Note that this is not available for Data Sources (such as IDF System-Link) that are guaranteed to return a unique key for every Data Object.

The Used By Detail shows a Where Used list of objects that are currently using this Property.



Domain Object Property

Data Object: Customer Account
Property Name: amount_due_30
Description: Amount Due (30)
Native Data Type: Decimal

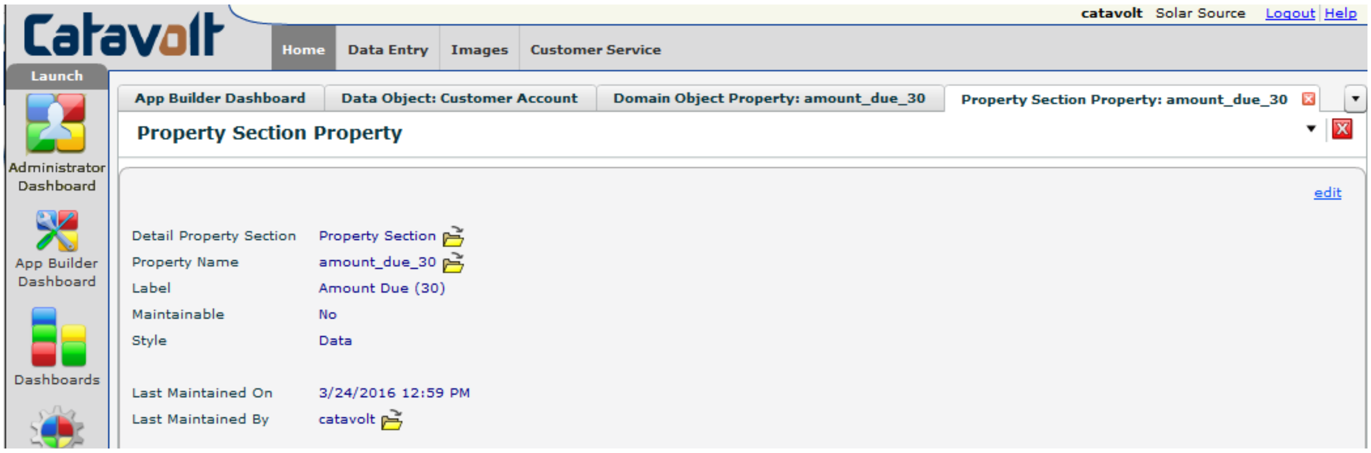
Last Maintained On: 3/24/2016 12:58 PM
Last Maintained By: catavolt

Where Used

Data Source 1	Data Object 2	Query 3	Detail 4	Action 5	Dashboard 6	BI Data Set 7	BI Dash 8	BI Dash Section 9	Type 10	Name 11	Las
SolarSource	Customer Account		Details						Detail Section Property	amount_due_30	3/2

Figure 36: Property 'Used By' details

Opening a Where Used record will take you directly to the object using the Property:


Catavolt
catavolt Solar Source Logout | Help
Home
Data Entry
Images
Customer Service
Launch
Administrator Dashboard
App Builder Dashboard
Dashboards
App Builder Dashboard
Data Object: Customer Account
Domain Object Property: amount_due_30
Property Section Property: amount_due_30
edit

Property Section Property

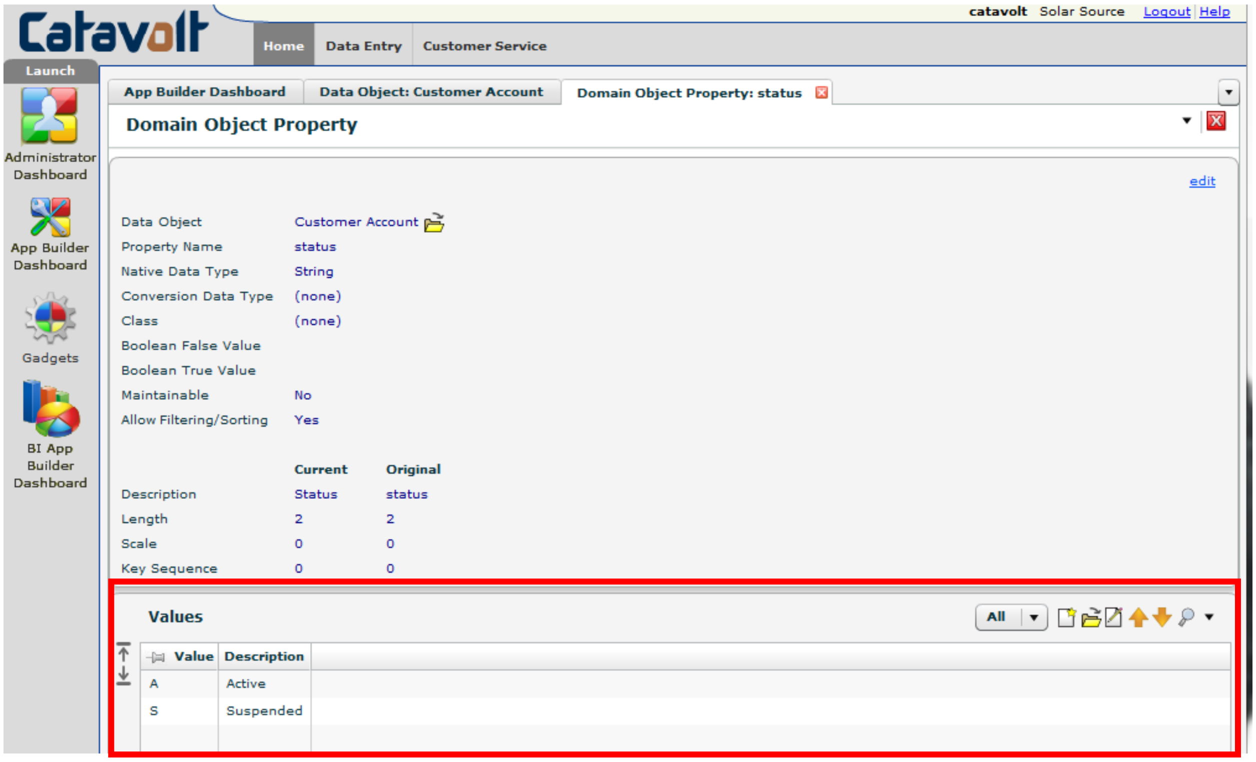
Detail Property Section: Property Section
Property Name: amount_due_30
Label: Amount Due (30)
Maintainable: No
Style: Data

Last Maintained On: 3/24/2016 12:59 PM
Last Maintained By: catavolt

Figure 37: Property Where Used

Defined Property Values

Certain properties may only have a specific set of allowable values. For example, the status property for a Customer Account may only allow an A (Active) or S (Suspended) to be entered as a value. You can control the list of available property values on the Values section of a Domain Object Property.



Domain Object Property

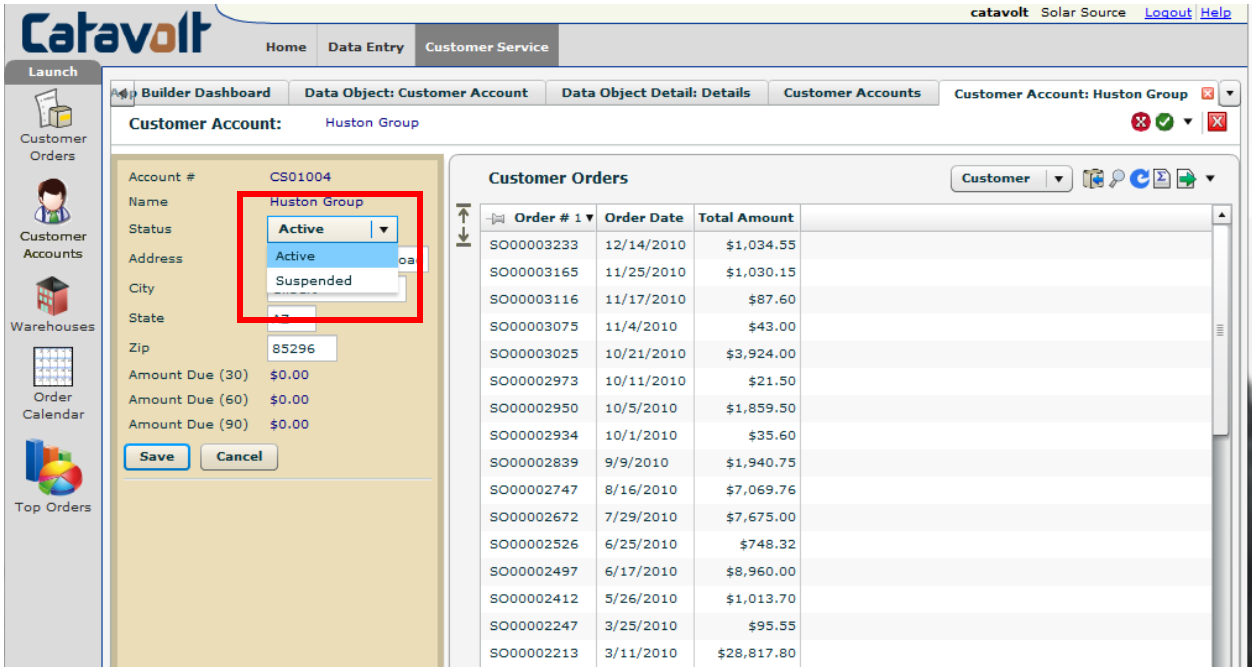
Data Object	Customer Account	
Property Name	status	
Native Data Type	String	
Conversion Data Type	(none)	
Class	(none)	
Boolean False Value		
Boolean True Value		
Maintainable	No	
Allow Filtering/Sorting	Yes	
	Current	**Original**
Description	Status	status
Length	2	2
Scale	0	0
Key Sequence	0	0

Values

Value	Description
A	Active
S	Suspended

Figure 38: Domain Object Property details with the Values query section highlighted.

By entering one or more Value records into this section Xalt will add a dropdown list to the property when it is displayed to allow the end user to select the appropriate value from a list. In addition, Xalt will show the value's description when displaying the property at runtime.



Customer Account: Huston Group

Account #	CS01004
Name	Huston Group
Status	Active ▼ Active Suspended
Address	
City	
State	
Zip	85296
Amount Due (30)	$0.00
Amount Due (60)	$0.00
Amount Due (90)	$0.00

Customer Orders

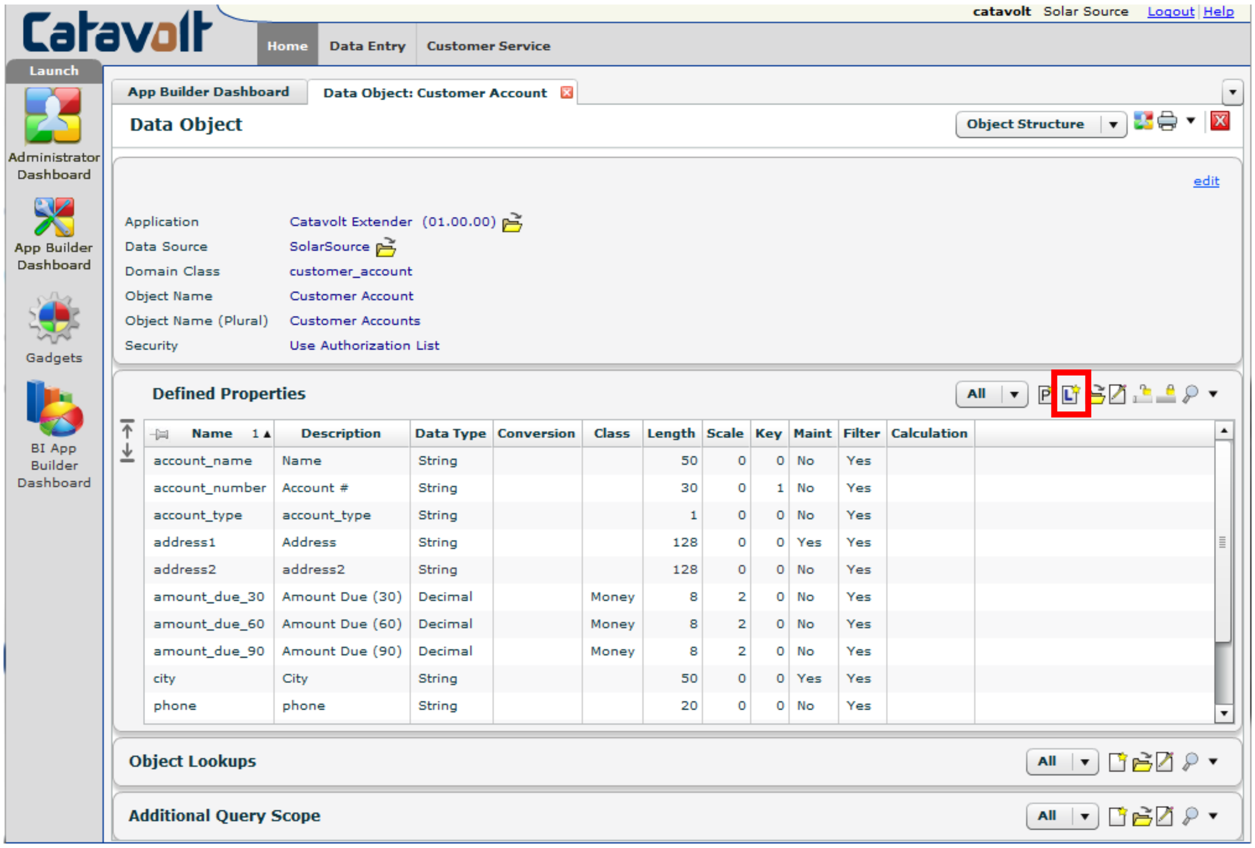
Order # 1 ▼	Order Date	Total Amount
SO00003233	12/14/2010	$1,034.55
SO00003165	11/25/2010	$1,030.15
SO00003116	11/17/2010	$87.60
SO00003075	11/4/2010	$43.00
SO00003025	10/21/2010	$3,924.00
SO00002973	10/11/2010	$21.50
SO00002950	10/5/2010	$1,859.50
SO00002934	10/1/2010	$35.60
SO00002839	9/9/2010	$1,940.75
SO00002747	8/16/2010	$7,069.76
SO00002672	7/29/2010	$7,675.00
SO00002526	6/25/2010	$748.32
SO00002497	6/17/2010	$8,960.00
SO00002412	5/26/2010	$1,013.70
SO00002247	3/25/2010	$95.55
SO00002213	3/11/2010	$28,817.80

Figure 39: Example Xalt runtime showing a drop down on a property.

Creating a Logical Property

Logical properties are properties on a Data Object that do not exist in the back-end system. Logical properties have their values derived based on an expression that combines constants with values of one or more physical fields. Note that some Data Source types (such as IDF System-Link) do not support logical properties.

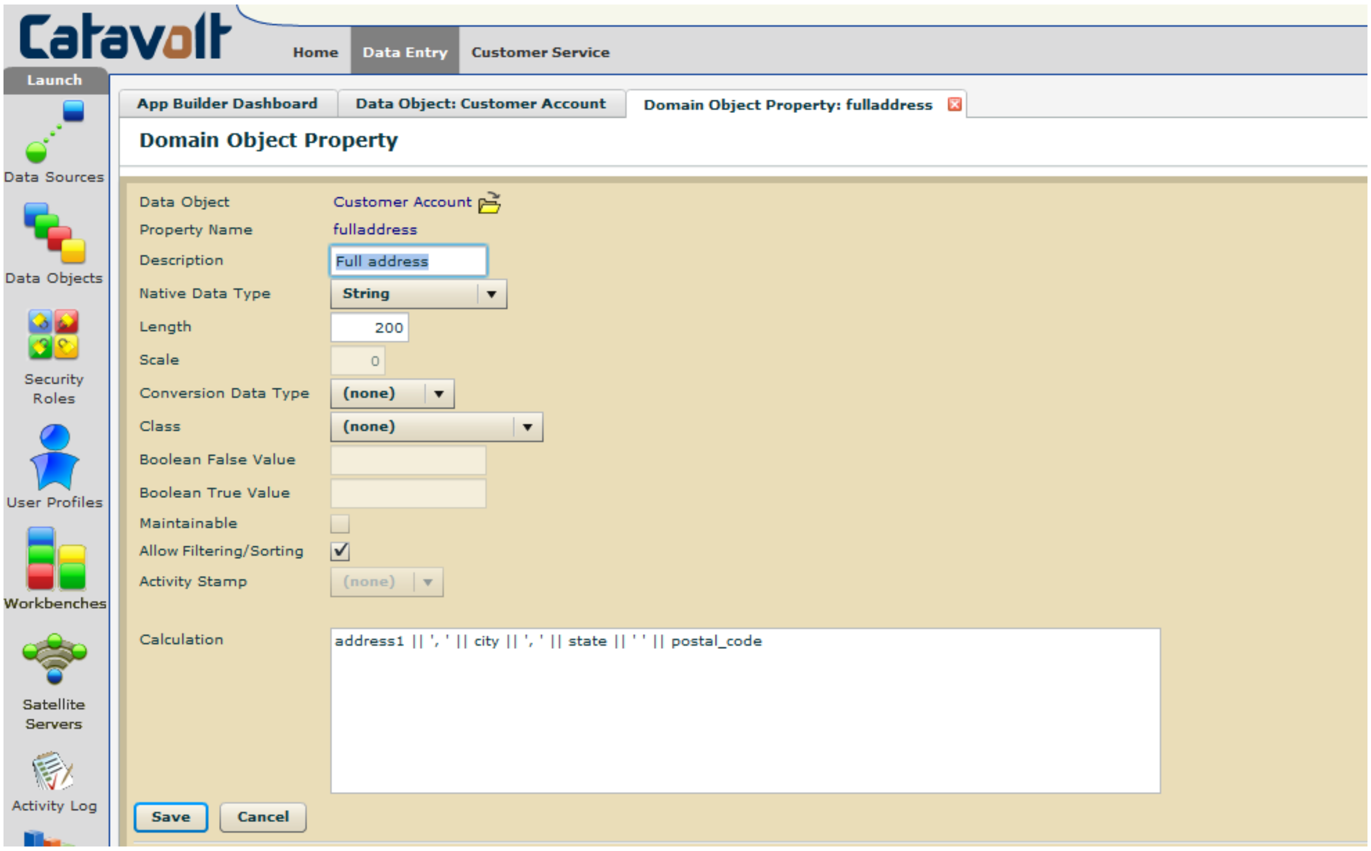
To create a logical property, select the Add Logical menu action:



Application: Catavolt Extender (01.00.00)
Data Source: SolarSource
Domain Class: customer_account
Object Name: Customer Account
Object Name (Plural): Customer Accounts
Security: Use Authorization List

Name 1 ▲	Description	Data Type	Conversion	Class	Length	Scale	Key	Maint	Filter	Calculation
account_name	Name	String			50	0	0	No	Yes	
account_number	Account #	String			30	0	1	No	Yes	
account_type	account_type	String			1	0	0	No	Yes	
address1	Address	String			128	0	0	Yes	Yes	
address2	address2	String			128	0	0	No	Yes	
amount_due_30	Amount Due (30)	Decimal		Money	8	2	0	No	Yes	
amount_due_60	Amount Due (60)	Decimal		Money	8	2	0	No	Yes	
amount_due_90	Amount Due (90)	Decimal		Money	8	2	0	No	Yes	
city	City	String			50	0	0	Yes	Yes	
phone	phone	String			20	0	0	No	Yes	

Figure 40: The Object Structure detail of a Data Object with the Add Logical action highlighted



Domain Object Property

Data Object	Customer Account
Property Name	fulladdress
Description	Full address
Native Data Type	String
Length	200
Scale	0
Conversion Data Type	(none)
Class	(none)
Boolean False Value	
Boolean True Value	
Maintainable	☐
Allow Filtering/Sorting	☑
Activity Stamp	(none)
Calculation	address1 || ', ' || city || ', ' || state || ' ' || postal_code

Figure 41: The Create Logical Property details

Data Object specifies which Data Object this Property is defined on. This value cannot be changed.

Property Name specifies the name this Property is known by. The value entered must be alphanumeric (value must begin with a letter (A-Z, a-z) and contain only letters and digits (A-Z,a-z,0-9)).

Description allows you to set a default description to be used when this property is added to a Query, Detail, Action, etc.

Native Data Type is the property type that Xalt should use to define this property. Allowable values are Boolean, Date, Decimal, Large Binary, Whole Number, String, Time, Timestamp. Large Binary specifies an Image or Signature capture from a Mobile Form. Logical Large Binary fields will be stored in Hexagon's cloud service.

Length specifies the property's length.

Scale specifies the number of decimals for a numeric property

Conversion Data Type is the property type as presented to the Xalt user. Note that this type can be different from the Native Data Type.

Decimal or Long types can have the following Conversion Data Types:

- Boolean – Presented to the Xalt User as a true/false switch. Stored in the back-end as a number specified by the Boolean True Value and Boolean False Value properties.
- Date (CYYMMDD) – Presented to the Xalt User as a Date field. Stored in the back--end system as a 7--digit value (C=Century (20th=0, 21st=1,etc), YY=2 digit Year, MM=Month, DD=Day)
- Date (YYYYMMDD) – Presented to the Xalt User as a Date field. Stored in the back--end system as an 8--digit value (YYYY=4 digit Year, MM=Month, DD=Day)
- Time (HHMMSS) – Presented to the Xalt User as a Time field. Stored in the back--end system as a 6--digit value (HH=Hour (24--hour format), MM=Minutes, SS=Seconds)

- Time (HHMM) – Presented to the Xalt User as a Time field. Stored in the back--end system as a 4--digit value (HH=Hour (24--hour format), MM=Minutes)

String types can have the following Conversion Data Types:

- Boolean – Presented to the Xalt User as a true/false switch. Stored in the back-end as a char specified by the Boolean True Value and Boolean False Value properties.

Timestamp types can have the following Conversion Data Types:

- Date – Presented to the Xalt User as a Date field. Stored in the back--end system as a Timestamp field with the Time portion set to 00:00:00.
- Time – Presented to the Xalt User as a Time field. Stored in the back--end system as a Timestamp field with the Date portion set to 1/1/1970.

Class specifies special formatting options used to display values for this property.

Decimal or Long types can have the following Classes:

- Barcode – Used to indicate a Barcode value
- Barcode (Auto-Scan) – Prevents a mobile device's soft keyboard from appearing on screen when this field gets normal focus. Manually selecting the field will subsequently bring up the soft keyboard if manual entry is required.
- Money – Displayed as a monetary value with a $.
- NFC – Used to indicate a value from an NFC tag
- Percent – Displayed as a percent value with percent sign.
- Unformatted Number – Displayed without thousands separators. This is typically used for non--quantity fields that are stored in the database as numeric fields (such as account numbers).
- Latitude – Used to indicate the latitude value for a location (used in Mapping).
- Longitude – Used to indicate the longitude value for a location (used in Mapping).

String types can have the following Classes:

- Barcode – Used to indicate a Barcode value
- Barcode (Auto-Scan) – Prevents a mobile device's soft keyboard from appearing on screen when this field gets normal focus. Manually selecting the field will subsequently bring up the soft keyboard if manual entry is required.
- E-mail – Used to indicate an e-mail address.
- HTML – Used to indicate that data values contain HTML tags which should be honored during rendering.
- NFC – Used to indicate a value from an NFC tag
- URL – Used to indicate a web URL.
- Password – Used to indicate a password. Password fields are display using ***** chars.
- Latitude/Longitude – Used to indicate the latitude/longitude value for a location (used in Mapping).
- Telephone – Used to indicate a Phone Number.
- Text – Used to indicate the String value should be rendered using a multi-line text control instead of a single-line entry field on details and actions.

Boolean False Value – For String and Numeric values that have a Conversion Data Type of Boolean, this indicates the actual value that should be read from / written to the back-end system for false property values. Note that this is not needed for native Boolean fields, as these fields already support native false values.

Boolean True Value - For String and Numeric values that have a Conversion Data Type of Boolean, this indicates the actual value that should be read from / written to the back-end system for true property values. Note that this is not needed for native Boolean fields, as these fields already support native true values.

Maintainable specifies whether the value for this property can be changed by the end user.

Allow Filtering/Sorting specifies whether the end user can use this property to change the filter and/or sort they are using for the current query. You may choose to turn this option off for properties with complex calculations that may cause performance issues when used for filtering or sorting.

Activity Stamp specifies a Date or Timestamp field used to indicate when Logical Large Binary properties have been uploaded. Since Logical Large Binary properties are stored in Hexagon's cloud service, your back-end system has no visibility into when these properties are set. Setting an Activity Stamp field will cause the property to have the current Date or current Date/Time filled in whenever the signature or image is uploaded. Removing an existing image will set the Activity Stamp field to null.

Calculation allows you to set an expression to calculate the value of this logical property. The expression must be defined in a format understandable by the back-end system, as Xalt will pass the expression directly to the back-end system to perform the calculation. Because of this, logical properties are only supported on Data Sources whose back-end can accept requests that contain calculations.

Other Defined Properties Menu Options

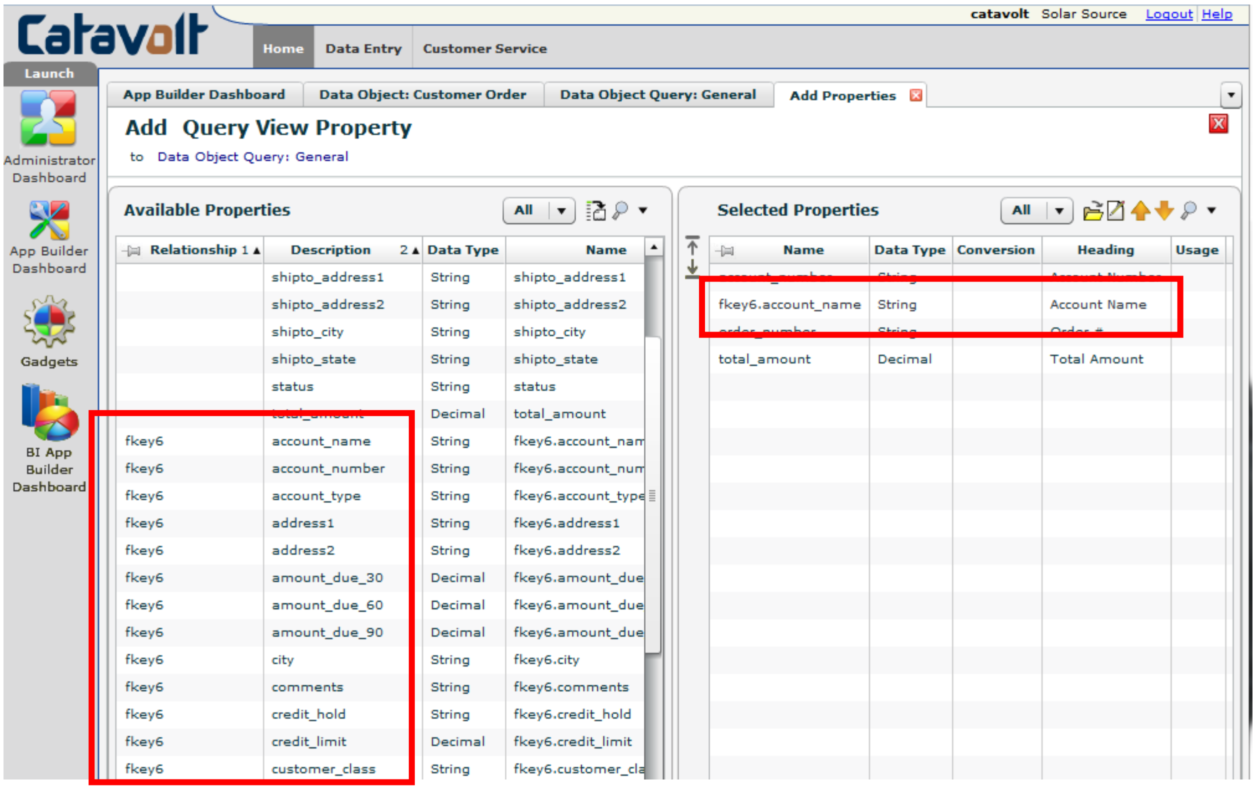
Reload Definition From Server

There are times where the definition of a property may change (e.g. due to updating the property's definition in an SQL Table or View). Selecting this menu option will go back to the server and update the Data Type, Length, and Scale for this property based on the current Server Meta Data values. In the past, you would need to run a full Refresh Meta Data from Server on the Data Object to fix this issue. Note that you should perform a Reset Cached Meta Data on the Data Source first to ensure that the new definition is pulled from the server.

Additional Query Scopes

A Query Scope is the list of all available properties from the back end system that can be used on Queries, Details, Actions, etc… Often times the Query Scope will span more than one back end entity.

If the database table that a Data Object is built over contains any Foreign Key definitions, properties for these foreign keys will automatically be loaded as available properties for the Data Object. You will be able to recognize these properties because the Relationship column on Available Properties is blank for local properties and non-blank for foreign key properties. When you refer to these properties, you use the format of <relationship>.<propertyName> .



Add Query View Property

to Data Object Query: General

Available Properties

Relationship 1 ▲	Description 2 ▲	Data Type	Name
	shipto_address1	String	shipto_address1
	shipto_address2	String	shipto_address2
	shipto_city	String	shipto_city
	shipto_state	String	shipto_state
	status	String	status
	total_amount	Decimal	total_amount
fkey6	account_name	String	fkey6.account_nam
fkey6	account_number	String	fkey6.account_num
fkey6	account_type	String	fkey6.account_type
fkey6	address1	String	fkey6.address1
fkey6	address2	String	fkey6.address2
fkey6	amount_due_30	Decimal	fkey6.amount_due
fkey6	amount_due_60	Decimal	fkey6.amount_due
fkey6	amount_due_90	Decimal	fkey6.amount_due
fkey6	city	String	fkey6.city
fkey6	comments	String	fkey6.comments
fkey6	credit_hold	String	fkey6.credit_hold
fkey6	credit_limit	Decimal	fkey6.credit_limit
fkey6	customer_class	String	fkey6.customer_cla

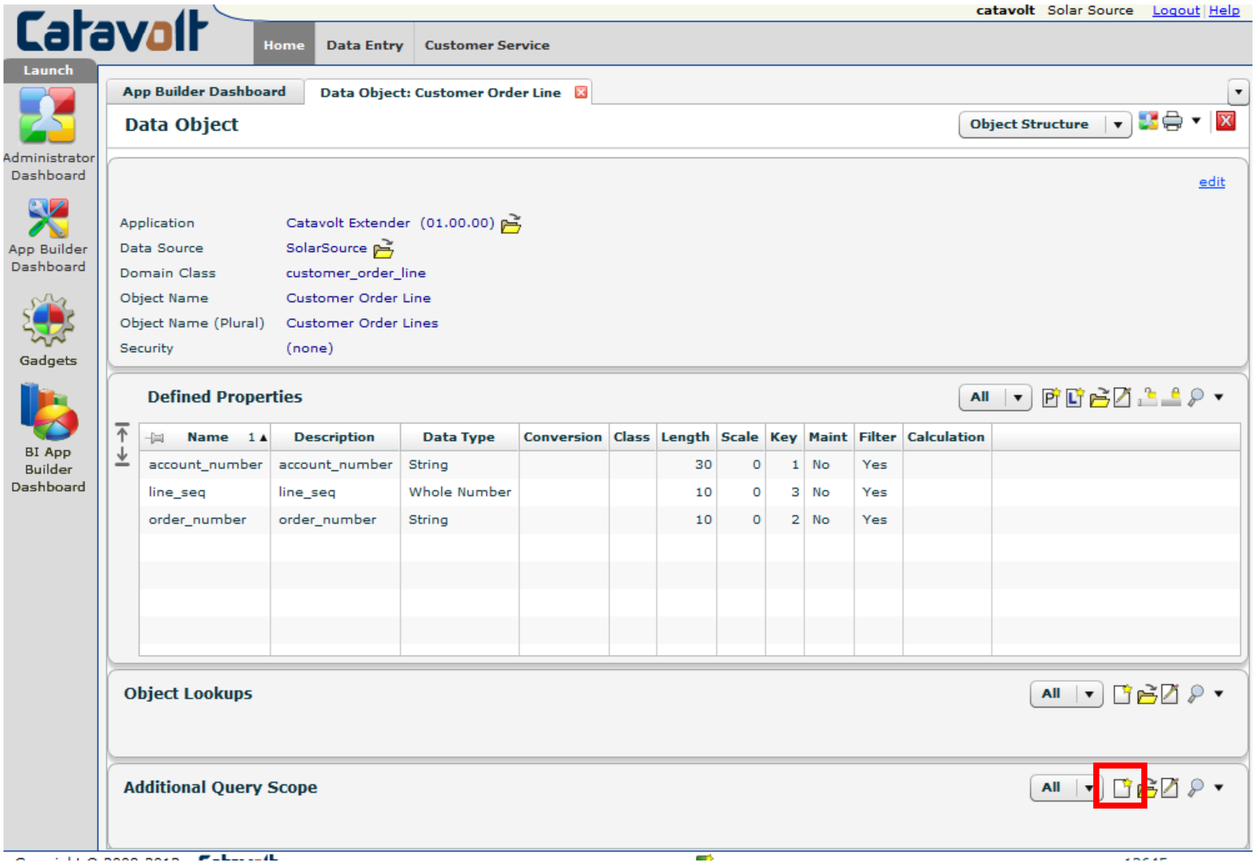
Selected Properties

Name	Data Type	Conversion	Heading	Usage
account_number	String		Account Number	
fkey6.account_name	String		Account Name	
order_number	String		Order #	
total_amount	Decimal		Total Amount	

Figure 42: Add Query View Property screen showing related properties from foreign key constraints.

For Data Objects that do not have foreign keys defined against the table, you many achieve the same result manually by adding an Additional Query Scope (note that due to the nature of how ODATA and IDF System-Link work, this feature is not available for Data Sources of these types. To achieve the same result for IDF System-Link, add a Many-1 relationship between the two objects in Enterprise Integrator).

To create an Additional Query Scope Entry, go to the Object Structure Detail view for the specified Data Object and press the Create button.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_order_line
Object Name	Customer Order Line
Object Name (Plural)	Customer Order Lines
Security	(none)

Defined Properties

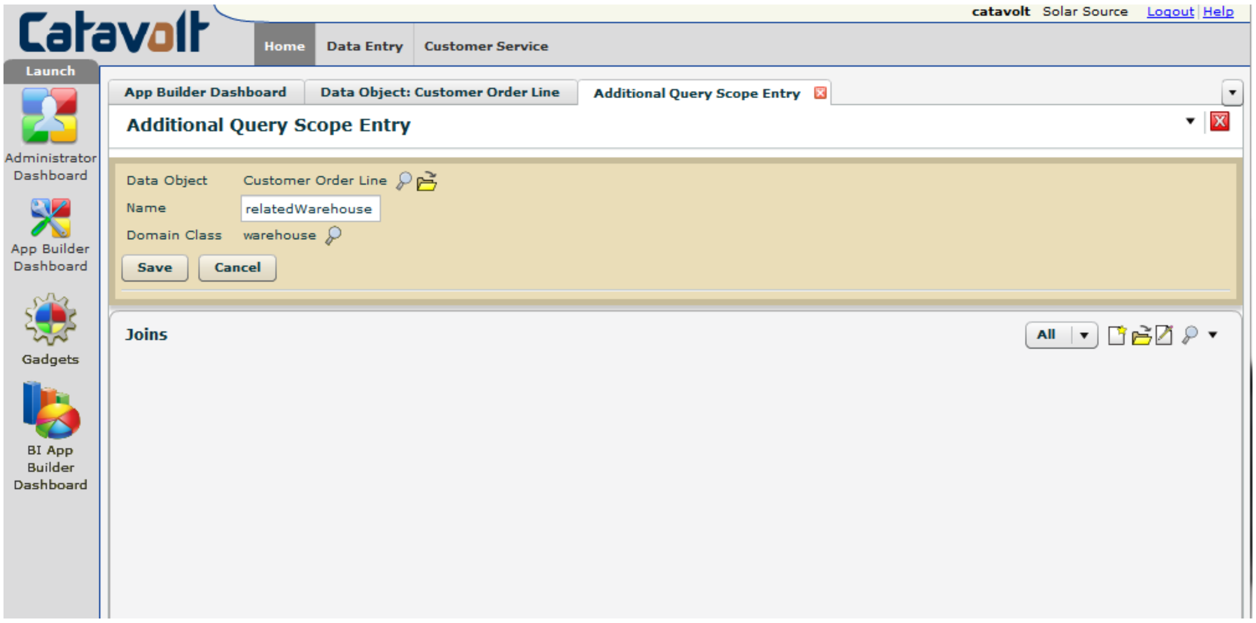
Name 1 ▲	Description	Data Type	Conversion	Class	Length	Scale	Key	Maint	Filter	Calculation
account_number	account_number	String			30	0	1	No	Yes	
line_seq	line_seq	Whole Number			10	0	3	No	Yes	
order_number	order_number	String			10	0	2	No	Yes	

Object Lookups

Additional Query Scope

Figure 43: A Data Object 'Object Structure' detail view showing the Additional Query Scope query section with the Create action highlighted.

Enter the name of the relationship and then press the Find button next to Domain Class. This will bring up a list of all the available tables or objects for your data source. Select the one you wish to use, then press the Save button.



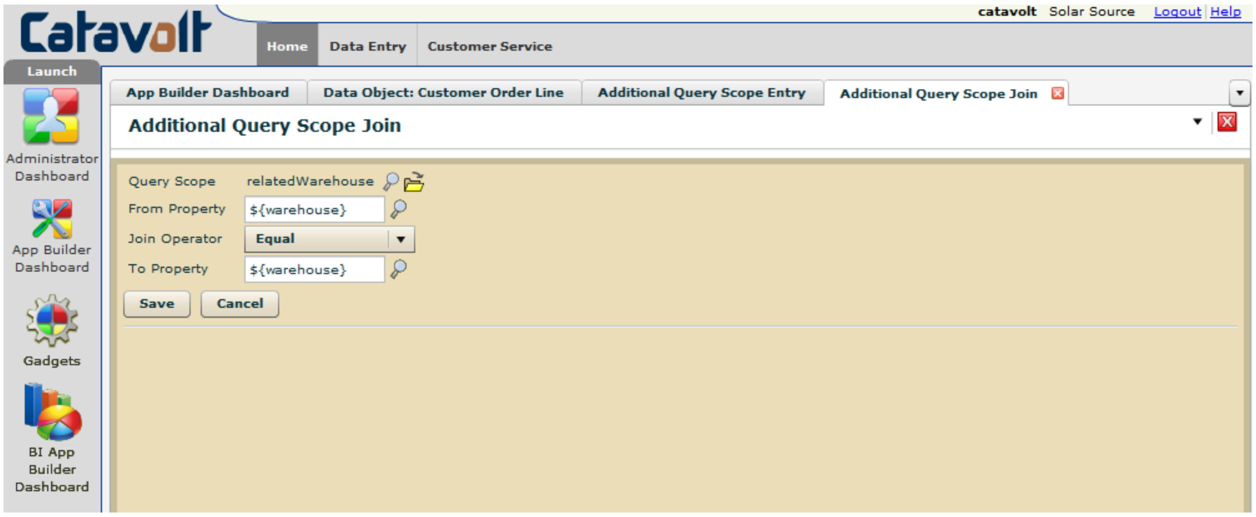
Additional Query Scope Entry

Data Object: Customer Order Line
Name: relatedWarehouse
Domain Class: warehouse

Save Cancel

Figure 44: The create Query Scope Entry view

Once you have identified the table you wish to join to, you must then define the Join criteria that will be used to actually connect the tables together. Press the create button in the Joins list section. In the most common cases, the join criteria is a series of field=field values (for example, customer=customer and company=company). Clicking the Find Property button next to From Property and To Property will show you a list of available properties from each table and allow you to choose which ones to use in the join. Xalt will then enter the property name using the substitution message format ${<property name>} to identify that a field from the table is being used. You may also choose to enter a constant value instead of a property name. Likewise, you have a list of operators to choose from in addition to Equals. See Appendix A: Specifying Messages and Substitution Values for more information about how to create a Substitution Message for From Property and To Property.



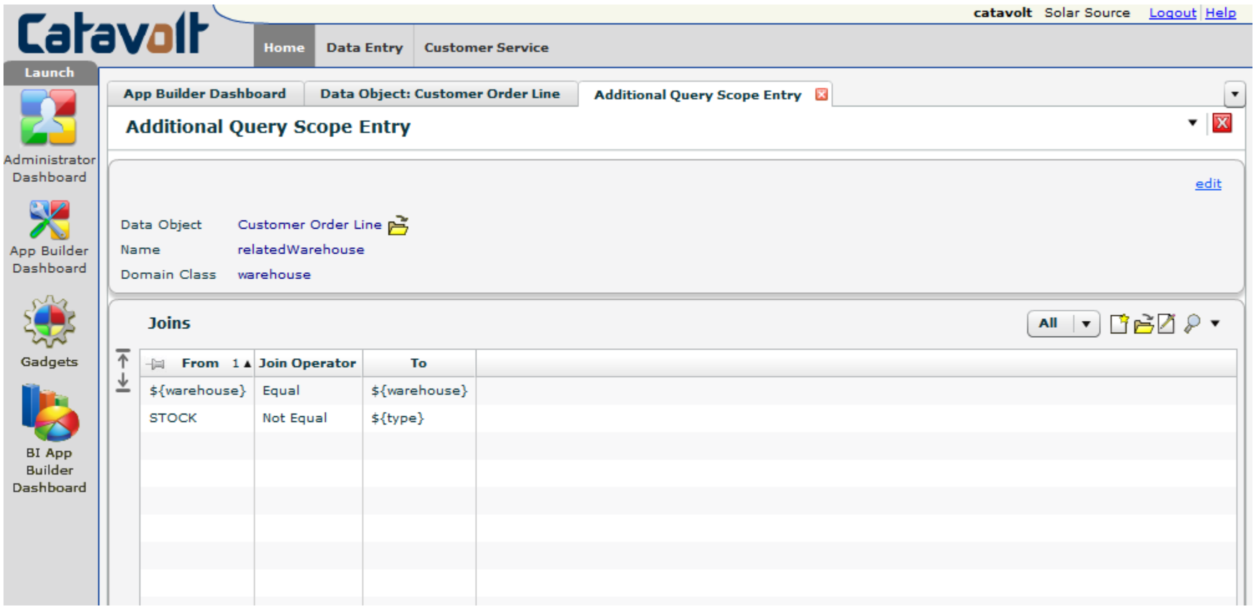
Additional Query Scope Join

Query Scope: relatedWarehouse
From Property: ${warehouse}
Join Operator: Equal
To Property: ${warehouse}

Save Cancel

Figure 45: The create Additional Query Scope Join view

As you can see below, you may enter in multiple joins to be used. Xalt will use all the Joins, ANDing them together when querying the data.



Additional Query Scope Entry

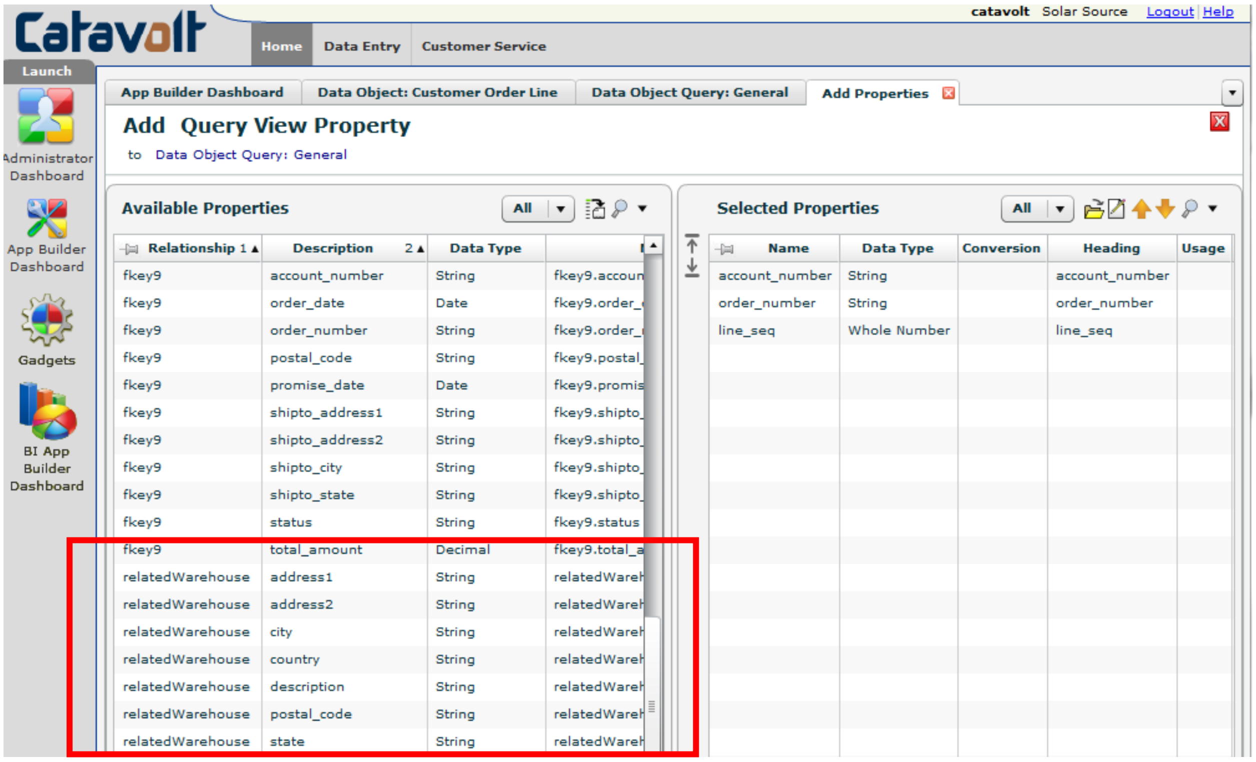
Data Object	Customer Order Line
Name	relatedWarehouse
Domain Class	warehouse

Joins

From 1	Join Operator	To
${warehouse}	Equal	${warehouse}
STOCK	Not Equal	${type}

Figure 46: An Additional Query Scope Entity example

When you have completed creating your Additional Query Scope entry, you will notice that fields from your joined-in table will now be available to be added to Queries, Details, etc.



Add Query View Property

to Data Object Query: General

Available Properties

Relationship 1	Description 2	Data Type	
fkey9	account_number	String	fkey9.accoun
fkey9	order_date	Date	fkey9.order_
fkey9	order_number	String	fkey9.order_
fkey9	postal_code	String	fkey9.postal_
fkey9	promise_date	Date	fkey9.promis
fkey9	shipto_address1	String	fkey9.shipto_
fkey9	shipto_address2	String	fkey9.shipto_
fkey9	shipto_city	String	fkey9.shipto_
fkey9	shipto_state	String	fkey9.shipto_
fkey9	status	String	fkey9.status
fkey9	total_amount	Decimal	fkey9.total_a
relatedWarehouse	address1	String	relatedWareh
relatedWarehouse	address2	String	relatedWareh
relatedWarehouse	city	String	relatedWareh
relatedWarehouse	country	String	relatedWareh
relatedWarehouse	description	String	relatedWareh
relatedWarehouse	postal_code	String	relatedWareh
relatedWarehouse	state	String	relatedWareh

Selected Properties

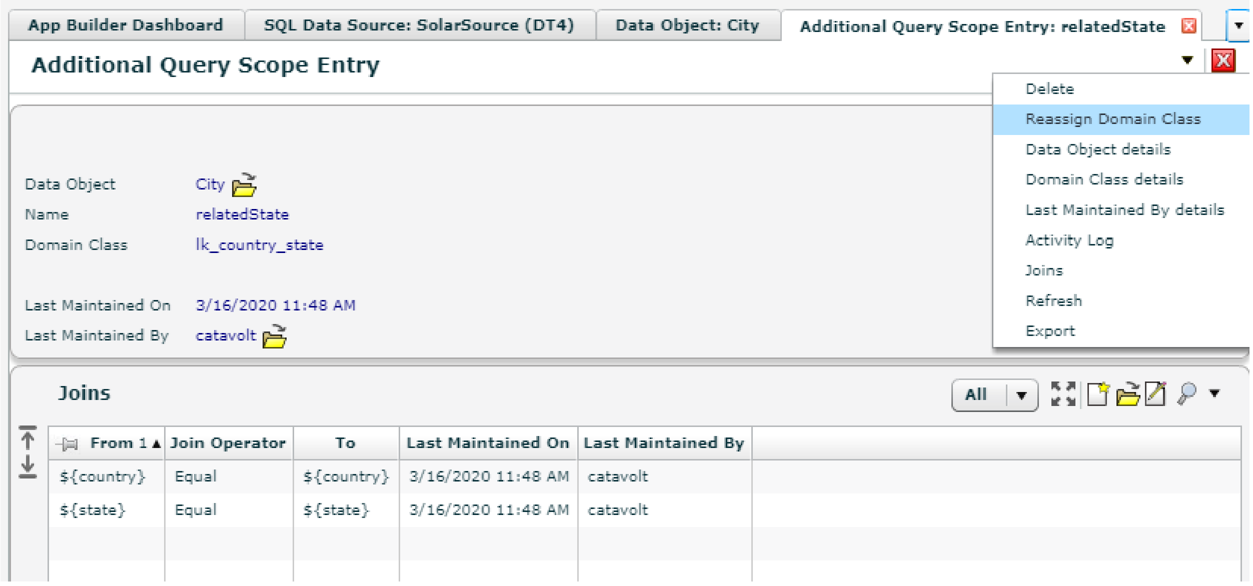
Name	Data Type	Conversion	Heading	Usage
account_number	String		account_number	
order_number	String		order_number	
line_seq	Whole Number		line_seq	

Figure 47: An Available Properties view showing properties from an Additional Query Scope named relatedWarehouse

Other Additional Query Scope Menu Options

Reassign Domain Class

Selecting this menu option will allow you to change the Domain Class at runtime. As this option can be destructive if not used properly, we have added it to the Details menu only. Additional Query Scope List menus will not have this menu option.



Additional Query Scope Entry

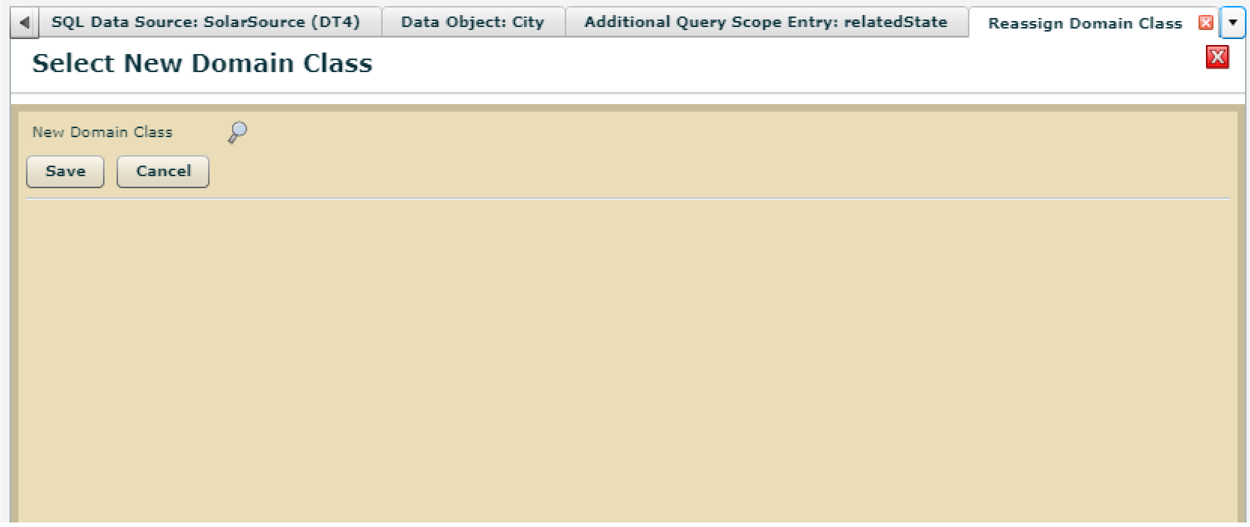
Data Object	City
Name	relatedState
Domain Class	lk_country_state
Last Maintained On	3/16/2020 11:48 AM
Last Maintained By	catavolt

Joins

From 1 ▲	Join Operator	To	Last Maintained On	Last Maintained By
${country}	Equal	${country}	3/16/2020 11:48 AM	catavolt
${state}	Equal	${state}	3/16/2020 11:48 AM	catavolt

Figure 48: An Additional Query Scope view showing the Reassign Domain Class menu action

Running this menu action will bring up a prompt with an option to choose a new Domain Class (the same as running the Reassign Domain Class menu option on Data Object):



SQL Data Source: SolarSource (DT4)
Data Object: City
Additional Query Scope Entry: relatedState
Reassign Domain Class
Save
Cancel

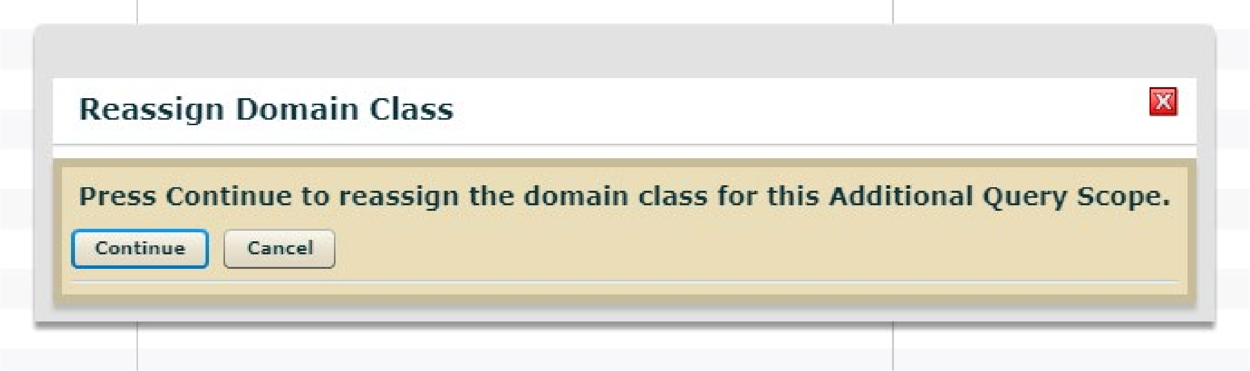
Select New Domain Class

New Domain Class

Figure 49: The Reassign Domain Class details view

When selecting a Domain Class, the server will run a number of checks to determine if any potential issues exist with renaming the Domain Class.

If no potential issues are found, a confirmation dialog is presented to complete domain class reassignment:

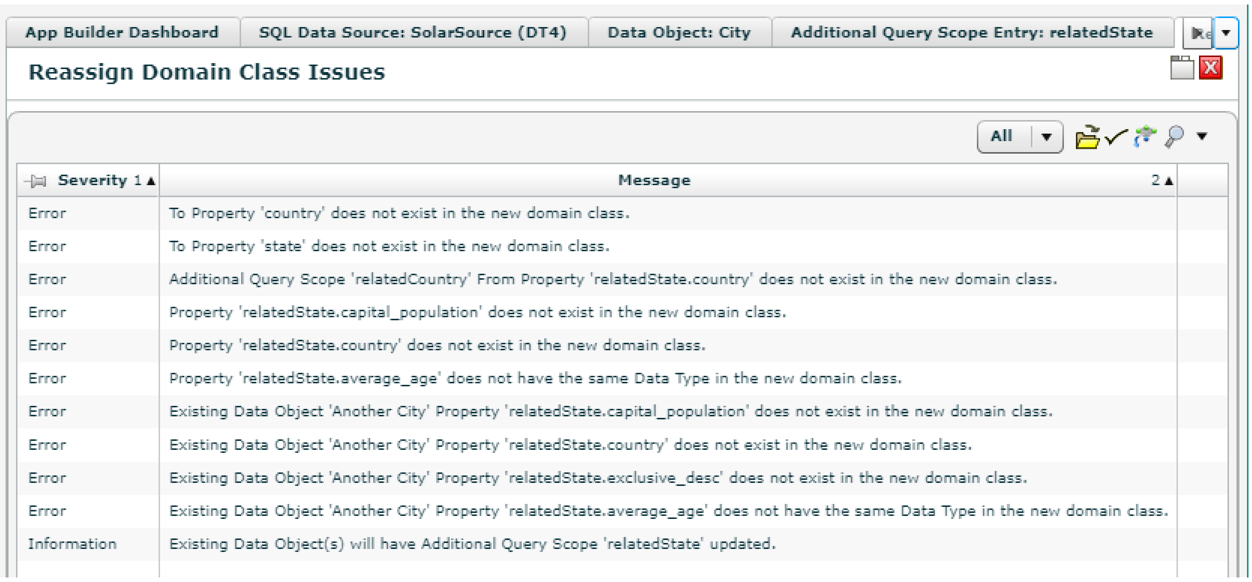


Reassign Domain Class

Press Continue to reassign the domain class for this Additional Query Scope.

Figure 50: The Reassign Domain Class confirmation prompt

If any potential issues are found, a list of the issues will be presented:


App Builder Dashboard
SQL Data Source: SolarSource (DT4)
Data Object: City
Additional Query Scope Entry: relatedState
Reassign Domain Class Issues
All

Severity 1 ▲	Message 2 ▲
Error	To Property 'country' does not exist in the new domain class.
Error	To Property 'state' does not exist in the new domain class.
Error	Additional Query Scope 'relatedCountry' From Property 'relatedState.country' does not exist in the new domain class.
Error	Property 'relatedState.capital_population' does not exist in the new domain class.
Error	Property 'relatedState.country' does not exist in the new domain class.
Error	Property 'relatedState.average_age' does not have the same Data Type in the new domain class.
Error	Existing Data Object 'Another City' Property 'relatedState.capital_population' does not exist in the new domain class.
Error	Existing Data Object 'Another City' Property 'relatedState.country' does not exist in the new domain class.
Error	Existing Data Object 'Another City' Property 'relatedState.exclusive_desc' does not exist in the new domain class.
Error	Existing Data Object 'Another City' Property 'relatedState.average_age' does not have the same Data Type in the new domain class.
Information	Existing Data Object(s) will have Additional Query Scope 'relatedState' updated.

Figure 51: The Reassign Domain Class Issues list view

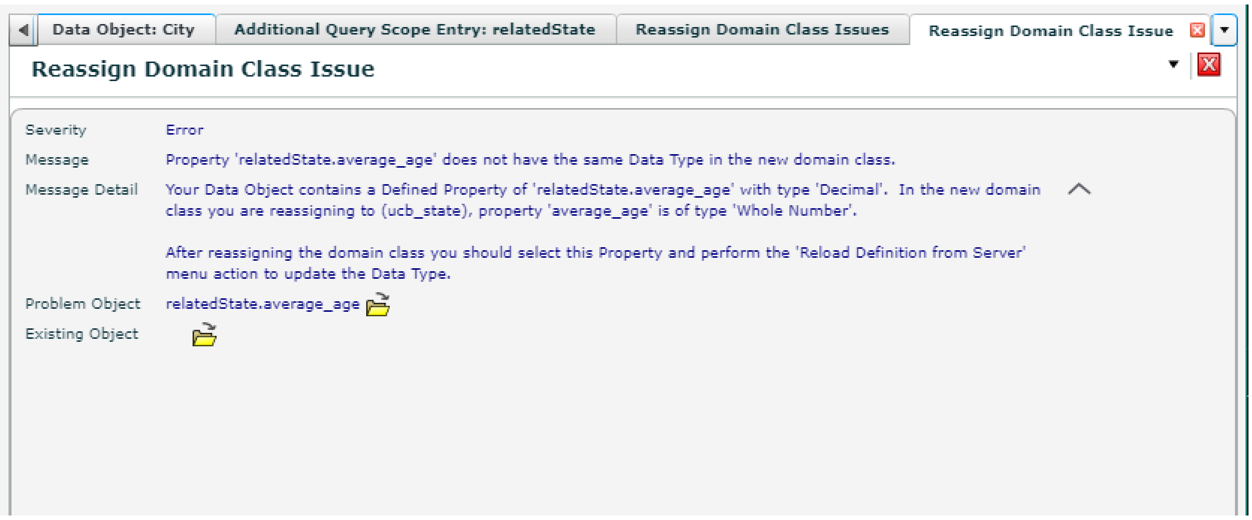
The list of potential issues to check for are:

1) A Defined Property for the Additional Query Scope (e.g relatedItem.xxx) does not exist in the new domain class
2) All Properties in 1) should have the same Data Type in the new domain class
3) An Additional Query Scope “To” Property does not exist in the new domain class
4) Other Additional Query Scope using a “relatedItem.xxx” “From” Property does not exist in the new domain class

Furthermore, if there are other Data Objects defined over the same domain class, the following issues may exist:

5) Issue 1) needs to be rechecked for all other Data Objects over the same domain class
6) Issue 2) needs to be rechecked for all other Data Objects over the same domain class

You can double-click any record to get more information about the issue:

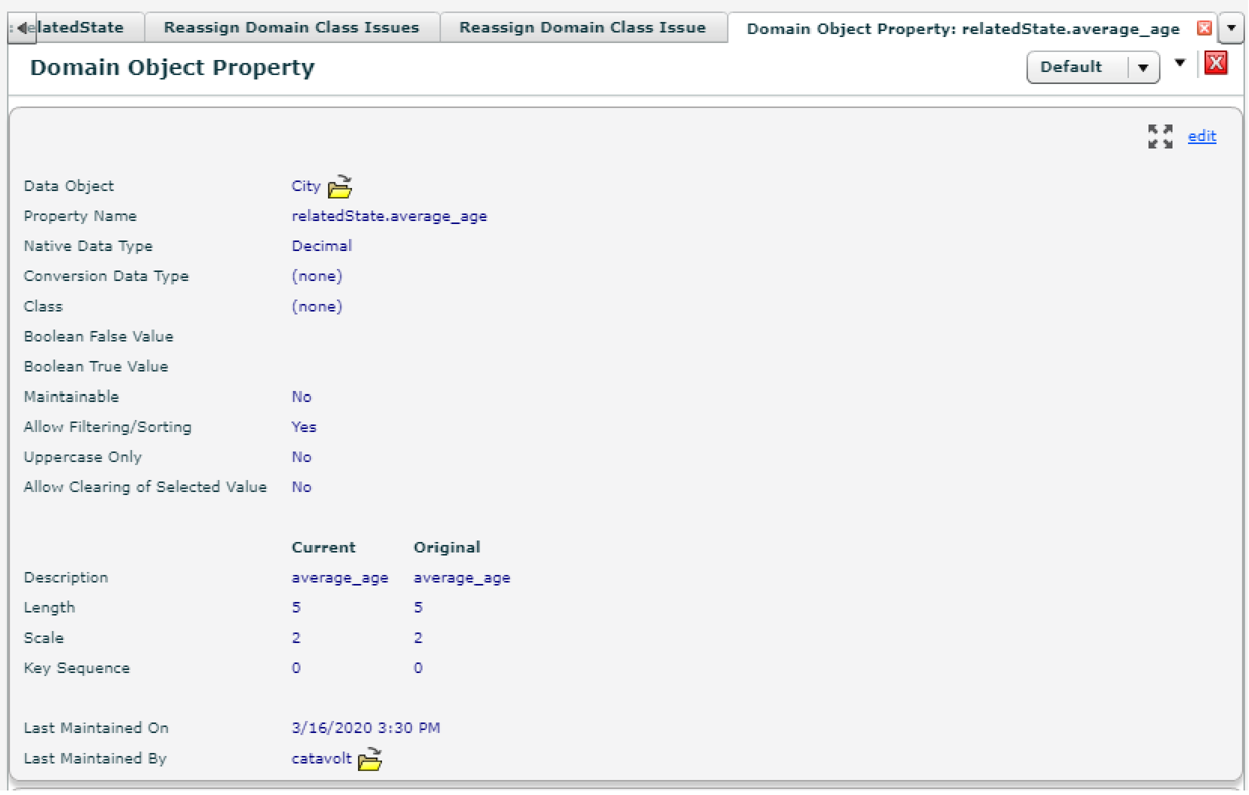

Data Object: City
Additional Query Scope Entry: relatedState
Reassign Domain Class Issues
Reassign Domain Class Issue

Reassign Domain Class Issue

Severity	Error
Message	Property 'relatedState.average_age' does not have the same Data Type in the new domain class.
Message Detail	Your Data Object contains a Defined Property of 'relatedState.average_age' with type 'Decimal'. In the new domain class you are reassigning to (ucb_state), property 'average_age' is of type 'Whole Number'. After reassigning the domain class you should select this Property and perform the 'Reload Definition from Server' menu action to update the Data Type.
Problem Object	relatedState.average_age
Existing Object	

Figure 52: The Reassign Domain Class Issues detail view

The Issue details will contain more detailed information about the potential issue and what should be done to resolve it. In addition, a direct link to the problem object is available. Clicking the open button next to it will bring up the object directly for review:


latedState
Reassign Domain Class Issues
Reassign Domain Class Issue
Domain Object Property: relatedState.average_age
Default
edit

Domain Object Property

Data Object	City	
Property Name	relatedState.average_age	
Native Data Type	Decimal	
Conversion Data Type	(none)	
Class	(none)	
Boolean False Value		
Boolean True Value		
Maintainable	No	
Allow Filtering/Sorting	Yes	
Uppercase Only	No	
Allow Clearing of Selected Value	No	
	Current	**Original**
Description	average_age	average_age
Length	5	5
Scale	2	2
Key Sequence	0	0
Last Maintained On	3/16/2020 3:30 PM	
Last Maintained By	catavolt	

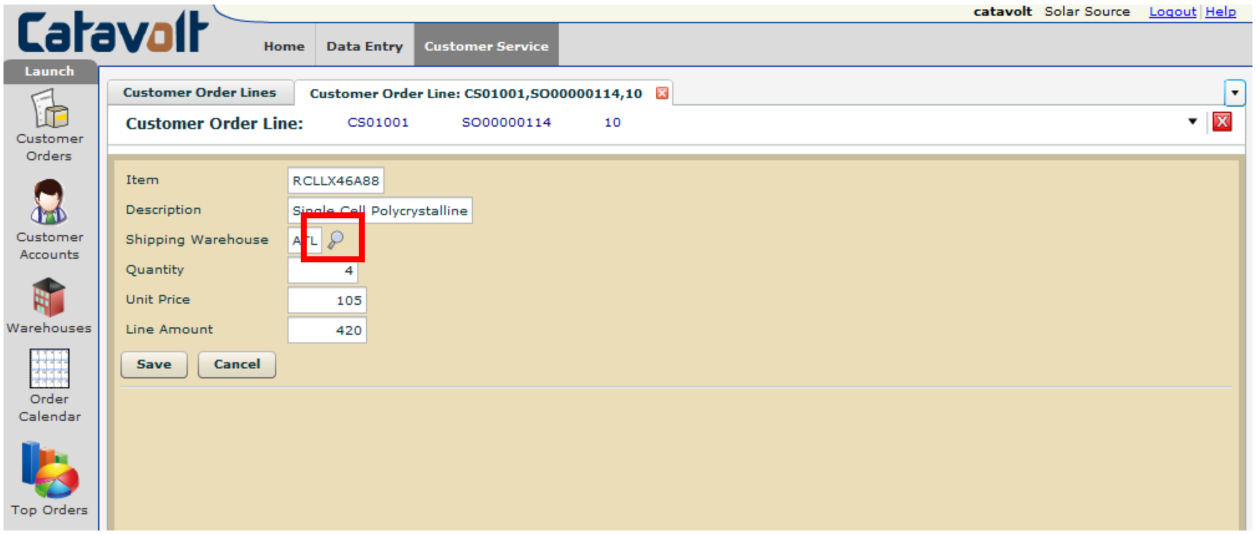
Figure 53: The result of running the Problem Object open action on the Reassign Domain Class Issues list view

As with Data Object, you can choose to manually resolve any of the above issues before the reassignment. At any point, you should be able to run an action from the issues list to complete the domain class reassignment, regardless of whether the issues have been resolved or not. After completing the reassignment, the issues list will remain open for you to use to investigate/resolve any remaining issues you did not resolve before the reassignment.

The reassignment process will simply change the domain class name to match your choice. If no other Data Objects in the system exist over the new domain class, the process is complete. If existing Data Objects do exist over the new domain class name, the domain class will be changed on all corresponding Additional Query Scopes as well. This is because all Data Objects over the same domain class share the same Object Structure, which includes Logical Properties, Additional Query Scopes, and Key information.

Object Lookups

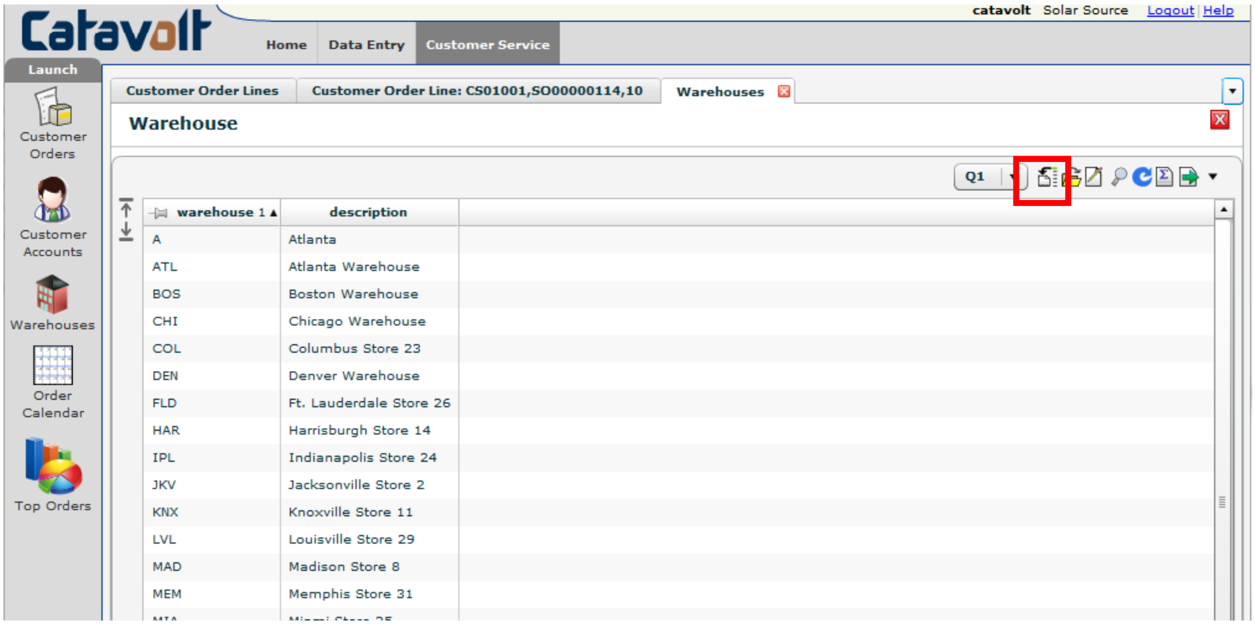
You may have situations where you have one Data Object that refers to another (for instance, a Customer Order Line refers to a Shipping Warehouse). Instead of manually typing in the warehouse for each order line, you would like to allow the user to press a button and select from a list of Warehouses. When the lookup button is pressed, a list query view for the lookup object is displayed. On this query view there is a menu and toolbar action for selecting the record the user would like to use. In addition, double clicking the record will also select the record. When a record is selected the user is returned to the originating screen and the property is filled in with the selected record.



Customer Order Line: CS01001 SO00000114 10

Item RCLLX46A88
Description Single Cell Polycrystalline
Shipping Warehouse ATL
Quantity 4
Unit Price 105
Line Amount 420

Figure 54: Xalt runtime example showing a warehouse field with a property lookup.

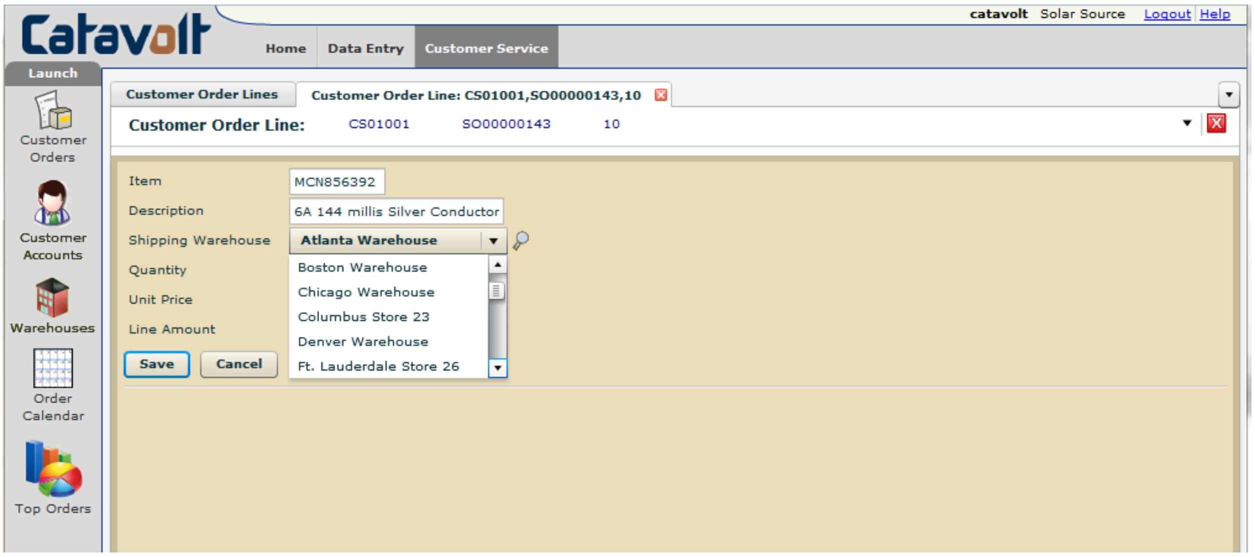


Warehouse

warehouse 1 ▲	description
A	Atlanta
ATL	Atlanta Warehouse
BOS	Boston Warehouse
CHI	Chicago Warehouse
COL	Columbus Store 23
DEN	Denver Warehouse
FLD	Ft. Lauderdale Store 26
HAR	Harrisburgh Store 14
IPL	Indianapolis Store 24
JKV	Jacksonville Store 2
KNX	Knoxville Store 11
LVL	Louisville Store 29
MAD	Madison Store 8
MEM	Memphis Store 31
MIA	Miami Store 25

Figure 55: A find warehouse query view with the Select action highlighted.

In addition to being able to look up a value, you may also decide put the available values into a list to pick from:



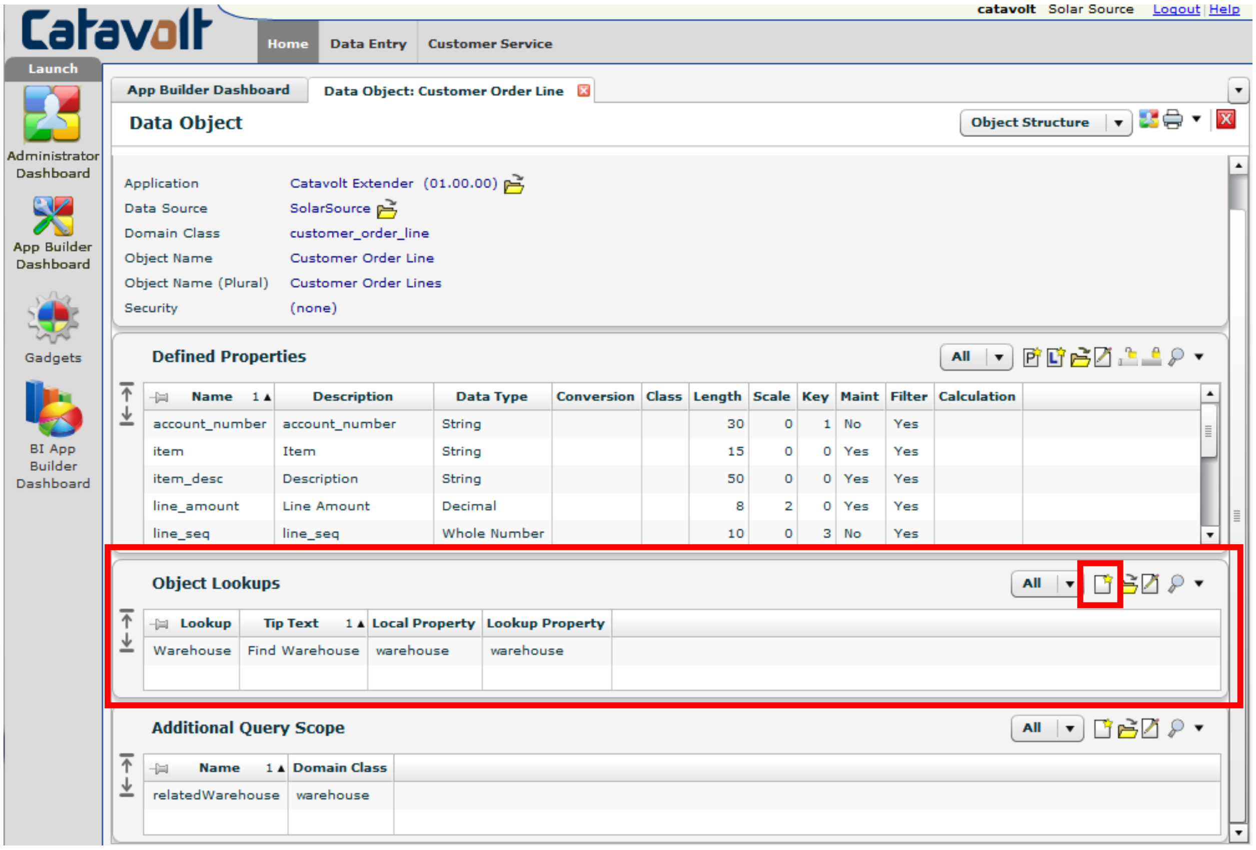
Customer Order Line: CS01001 SO00000143 10

Item	MCN856392
Description	6A 144 millis Silver Conductor
Shipping Warehouse	Atlanta Warehouse
Quantity	
Unit Price	
Line Amount	

Boston Warehouse
Chicago Warehouse
Columbus Store 23
Denver Warehouse
Ft. Lauderdale Store 26

Figure 56: The shipping warehouse field with a lookup and drop down list.

To create an Object Lookup, go to the Object Structure Detail view for the specified Data Object and press the Create button.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_order_line
Object Name	Customer Order Line
Object Name (Plural)	Customer Order Lines
Security	(none)

Defined Properties

Name 1 ▲	Description	Data Type	Conversion	Class	Length	Scale	Key	Maint	Filter	Calculation
account_number	account_number	String			30	0	1	No	Yes	
item	Item	String			15	0	0	Yes	Yes	
item_desc	Description	String			50	0	0	Yes	Yes	
line_amount	Line Amount	Decimal			8	2	0	Yes	Yes	
line_seq	line_seq	Whole Number			10	0	3	No	Yes	

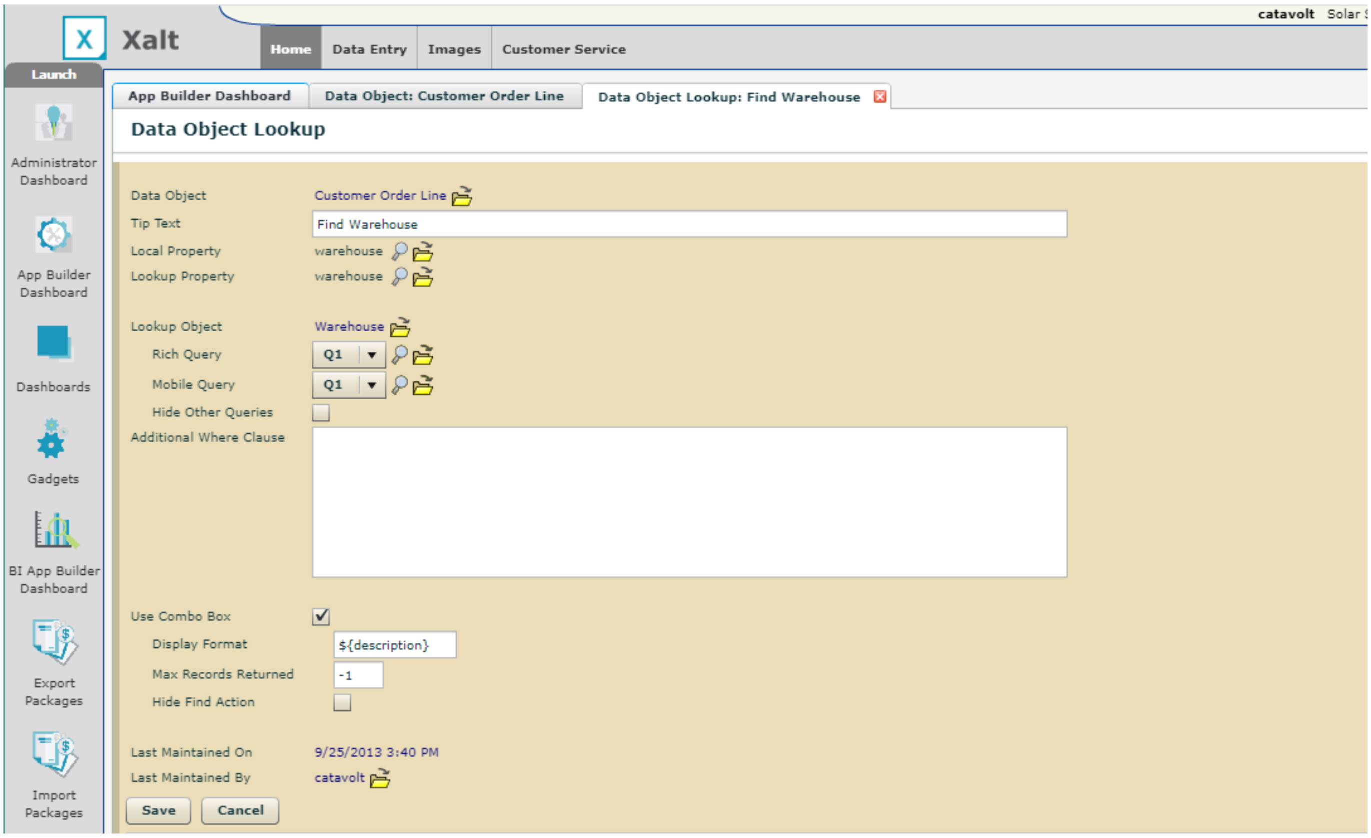
Object Lookups

Lookup	Tip Text 1 ▲	Local Property	Lookup Property
Warehouse	Find Warehouse	warehouse	warehouse

Additional Query Scope

Name 1 ▲	Domain Class
relatedWarehouse	warehouse

Figure 57: The Data Object Detail showing the Object Lookups query section with the Create action highlighted.



Data Object Lookup

Data Object	Customer Order Line
Tip Text	Find Warehouse
Local Property	warehouse
Lookup Property	warehouse
Lookup Object	Warehouse
Rich Query	Q1
Mobile Query	Q1
Hide Other Queries	□
Additional Where Clause	
Use Combo Box	☑
Display Format	${description}
Max Records Returned	-1
Hide Find Action	□
Last Maintained On	9/25/2013 3:40 PM
Last Maintained By	catavolt

Figure 58: The create Domain Object Lookup screen

First, you must select the Local Property. This specifies which property gets the find button and (optionally) the dropdown list. In our example, this is the Customer Property of the Order Data Object.

Next, you must select the Lookup Object. This is the Data Object that the find button will show a list for the end user to select from. In our example, we will be looking up a list of Customers.

The Rich and Mobile Query fields specify which Query you want to use when displaying the list of Customers. When you choose the Lookup Object, the Rich and Mobile Query fields are automatically set to the default rich and mobile Queries defined for the Data Object.

Hide Other Queries specifies whether the runtime Query dropdown list will contain just the selected Rich/Mobile Query or all non-hidden Queries. If Hide Other Queries is selected, only the selected Rich/Mobile Query will be available to choose in the Query dropdown list.

The Tip Text field specifies the tip text to display for the find button. The Tip Text field is automatically set to "Find" plus the name of the Lookup Object.

Next, you must select the Lookup Property. This is the field in the Lookup Object whose value will be put into the Local Property as a result of the find. In our example, we will be taking the customerNumber property from Customer and putting it into the customer property from Order.

Optionally, you can specify an Additional Where Clause in cases where you want to filter the list of Lookup Objects to be a subset of the available records. This is also where you can set up "cascading" lookups, where selecting a record from one combo box will re-filter the contents of another combo box (see Cascading Object Lookups section below). See Appendix B: Specifying Where Clauses for more information about specifying Additional Where Clause values.

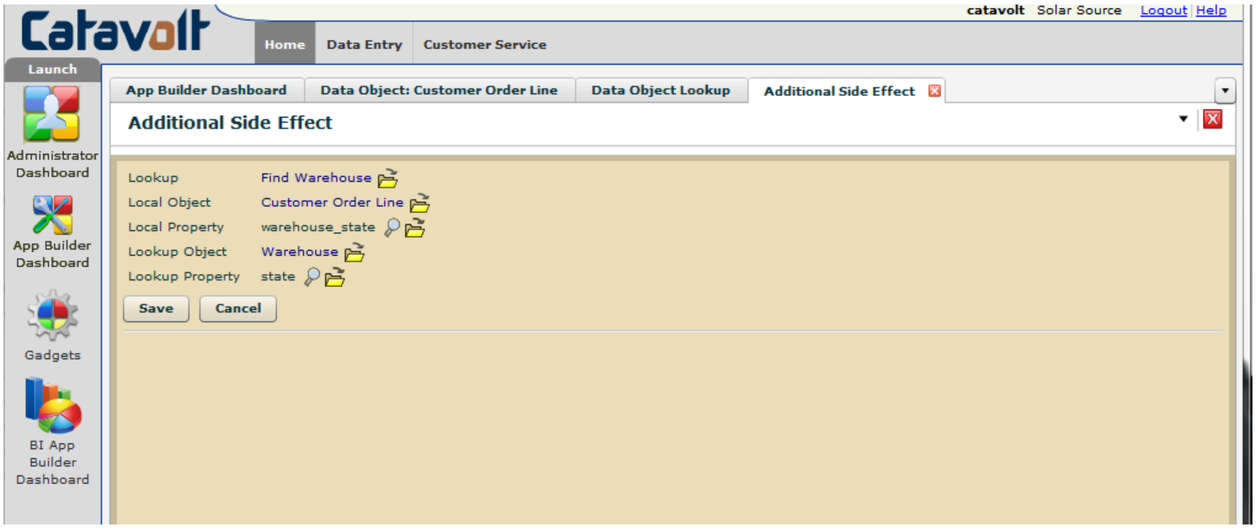
In some cases, you may want to allow users to choose from a list of available values instead requiring them to type in a value or perform an explicit lookup. To allow this behavior, select the Use Combo Box checkbox. The order in which the records will appear in the combo box is determined by the defined Sort of the Rich/Mobile Query being used.

Display Format specifies what text will appear in the combo box. Xalt defaults this value to the Lookup Object's Short Title. See Appendix A: Specifying Messages and Substitution Values for more options on specifying the Display Format.

Max Records Returned allows you return a specific number of records in the combo box. Xalt defaults this value to -1 (return all records). Using combo boxes for Lookup Objects with a large number of records can adversely affect performance. Setting the Max Records Returned to some reasonable value (in addition to using a sort that puts the most popular values first) can give the benefits of using combo boxes without degrading performance. For times when the value you need is not in the combo box list, you still have the ability to press the find button and get the full list of objects.

Hide Find Action allows you to hide the Find Toolbar action next to the combo box. This can be useful in cases where the Find Action provides no extra useful information over simply picking a value directly from the combo box.

The Additional Side Effects section allows you to set extra properties from the Lookup Object. This allows you to do an Object Lookup and return any number of properties back to your object.



Additional Side Effect

Lookup	Find Warehouse
Local Object	Customer Order Line
Local Property	warehouse_state
Lookup Object	Warehouse
Lookup Property	state

Figure 59: The create Additional Side Effect view

When creating an Additional Side Effect, you must select the Local Property. This specifies which property on the Data Object (in our example, the Order Data Object) will receive the returned value.

Next, you must select the Lookup Property. This is the field in the Lookup Object (in our example, the Customer Data Object) whose value will be put into the Local Property as a result of the find.

Cascading Object Lookups

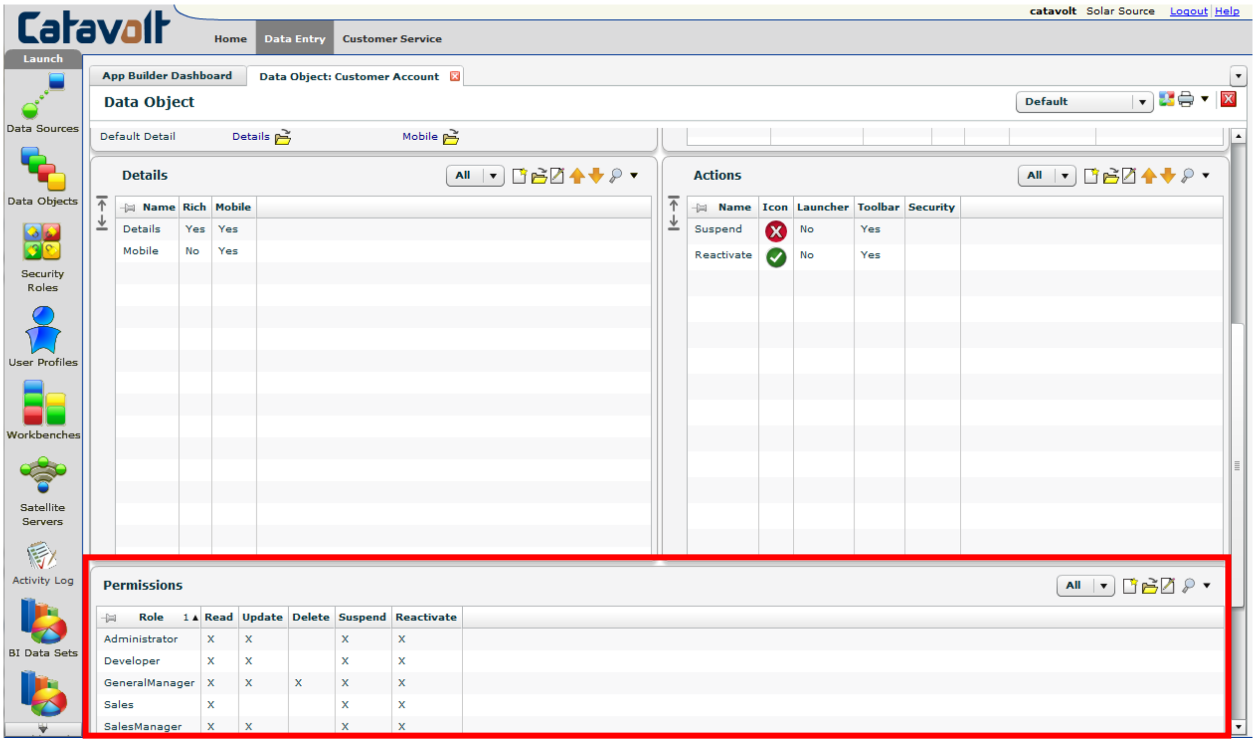
There are some instances where you may have multiple object lookups for a Data Object that are related to each other. A common example of this would be Country and State Data Objects. The Country object has a Country Code and Country Name. The State object has Country Code, State Code, and State Name, as a State exists within a Country. Now, using a normal Object Lookup, the State combo box will display all states for all countries. If you also have a Country field in your Data Object with an Object Lookup as well, you probably would like to "link" them together so that the State combo box only shows States for the currently selected Country. As the Country changes, you would like the list of available States to be changed as well. You can accomplish this using the Additional Where Clause field of the State's Object Lookup. Let's say for example you have a database table called addresses with fields country and state. Furthermore, you have a states table with fields countryCode and stateCode, and you have a countries table with field countryCode. Your Additional Where Clause for the State Object Lookup will be:

'${FROM_OBJECT.country}' = countryCode

This specifies that the value of the country field for Address must be equal to the countryCode field of State. Normally, Xalt will run this Additional Where Clause when it first brings up the Detail for the Data Object. However, Xalt has special behavior that will automatically reload a combo box whenever you select a value from a related combo box. In this way, as you choose a Country from its combo box, the list of States will automatically reload to only show States for the selected Country.

Permissions

If you set Security on your Data Object to Use Authorization List, you can use the Permissions section to define which Security Roles have authority to access and manipulate instances of these Data Objects. The Permissions section will show a list of Security Roles that have been granted authority to your Data Object, along with an X next to each action (standard action and user-defined action) that the role is authorized to perform.



Data Object

Details

Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

Actions

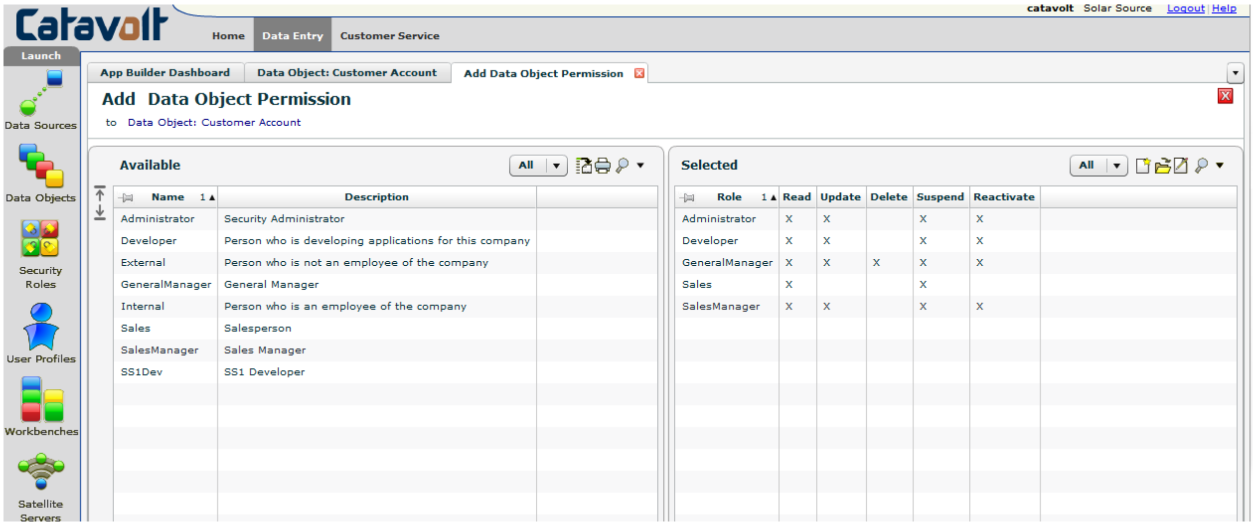
Name	Icon	Launcher	Toolbar	Security
Suspend	✗	No	Yes	
Reactivate	✓	No	Yes	

Permissions

Role 1 ▲	Read	Update	Delete	Suspend	Reactivate
Administrator	X	X		X	X
Developer	X	X		X	X
GeneralManager	X	X	X	X	X
Sales	X			X	X
SalesManager	X	X		X	X

Figure 60: Data Object details with the Permissions section highlighted

When adding Data Object Permissions, you will be presented with two lists. The Available Permissions list shows all Security Roles. The Selected Permissions list shows the roles that have been granted authority to this Data Object.



Add Data Object Permission

to Data Object: Customer Account

Available

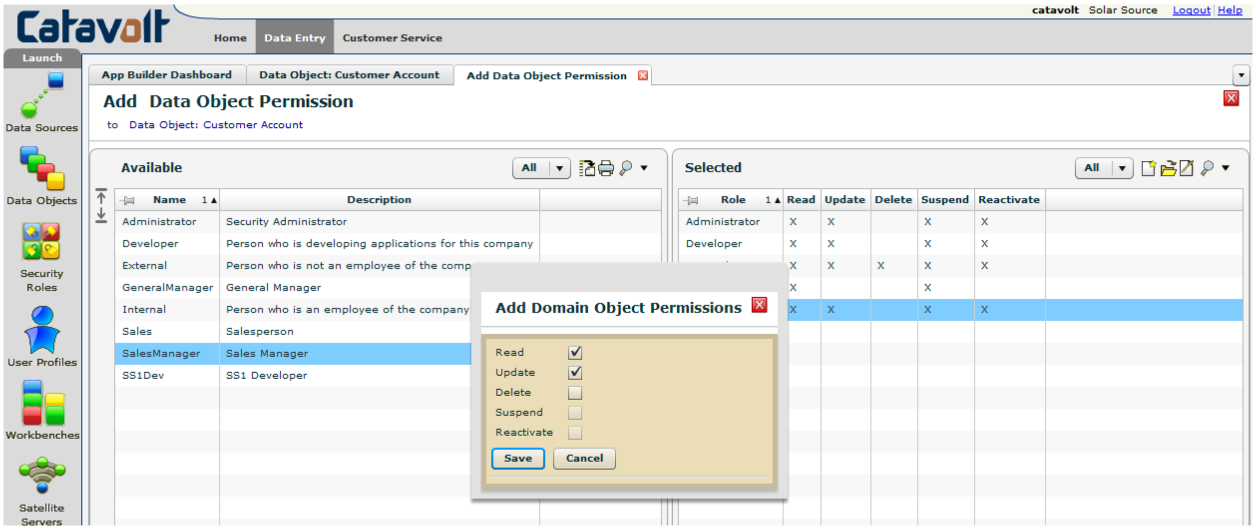
Name 1 ▲	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	SS1 Developer

Selected

Role 1 ▲	Read	Update	Delete	Suspend	Reactivate
Administrator	X	X		X	X
Developer	X	X		X	X
GeneralManager	X	X	X	X	X
Sales	X			X	
SalesManager	X	X		X	X

Figure 61: The add Data Object Permission view.

You can select a single or multiple Security Roles and press the Add button to add them to the Data Object. You will be presented with the following dialog:



Add Data Object Permission

to Data Object: Customer Account

Available

Name 1 ▲	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the comp
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	SS1 Developer

Selected

Role 1 ▲	Read	Update	Delete	Suspend	Reactivate
Administrator	x	x		x	x
Developer	x	x		x	x
	x	x	x	x	x
	x			x	
	x	x		x	x

Add Domain Object Permissions

Read ☑
Update ☑
Delete ☐
Suspend ☐
Reactivate ☐

Save Cancel

Figure 62: The add Data Object Permission dialog.

You will see options for Read, Update, and Delete, along with an option for each Action you have defined for this Data Object Check the boxes for the action you wish the Security Role to have authority to perform. Note that you can only check boxes for Actions that have Security of Use Authorization List.

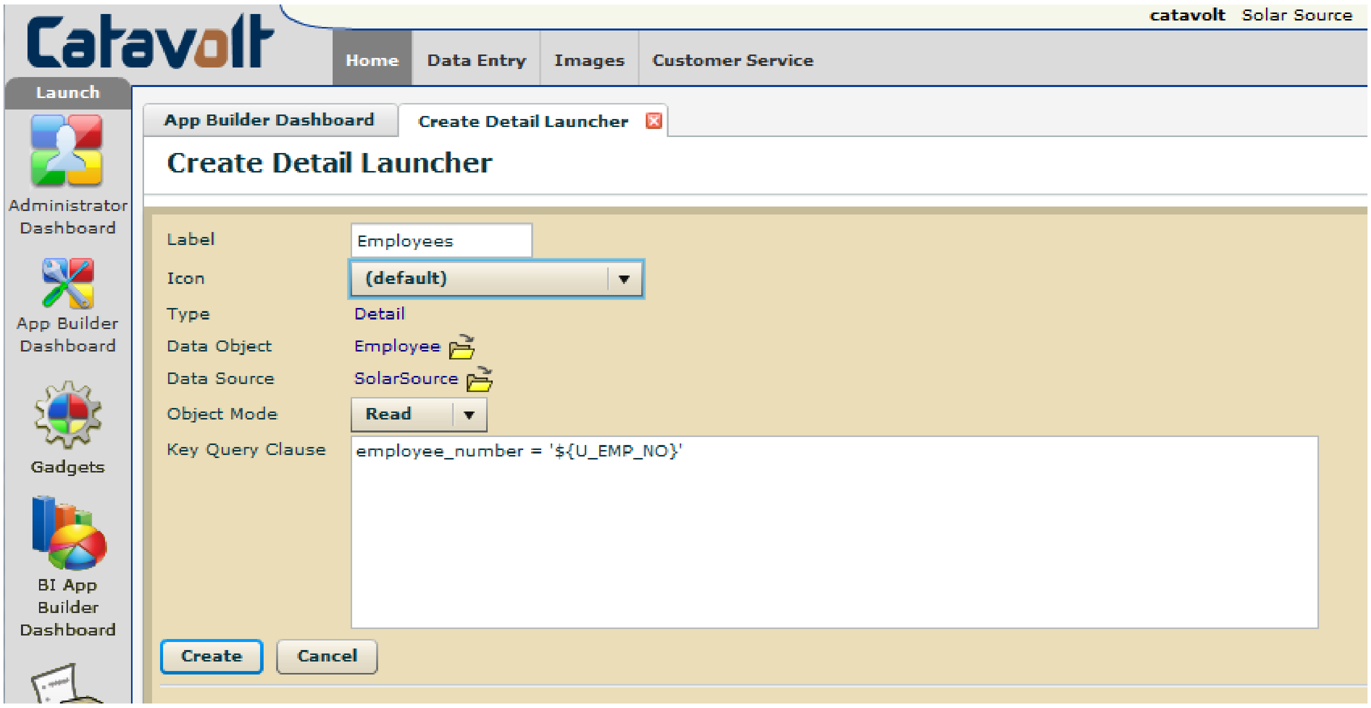
See Chapter 10: Security for more information about object security in Xalt.

Other Data Object Menu Options

Create Detail Launcher

In some cases, it is desirable to bring the end user directly to a single record from a launcher instead of the usual behavior of launching to a list of records. For example, on a Sales Rep Dashboard you may want to show the user a list of their Customers, Open Opportunities, Current Tasks, etc. Using the typical method, they would click on the Employee Launcher and get a list of Employees. They would then scroll down the list to find their record and then open their record to get to the work list. By creating a Detail Launcher, the end user can go directly to their record and bypass the initial list.

Use the Create Detail Launcher action on the Data Object toolbar or context menu to create a detail launcher. To perform this action, you must have a single Data Object selected. On the Create Detail Launcher view you will be asked to provide the following properties.



Create Detail Launcher

Label	Employees
Icon	(default)
Type	Detail
Data Object	Employee
Data Source	SolarSource
Object Mode	Read
Key Query Clause	employee_number = '${U_EMP_NO}'

Figure 63: The Create Detail Launcher view

Label specifies the text that will appear on the Launcher.

Icon specifies the image that will appear on the Launcher.

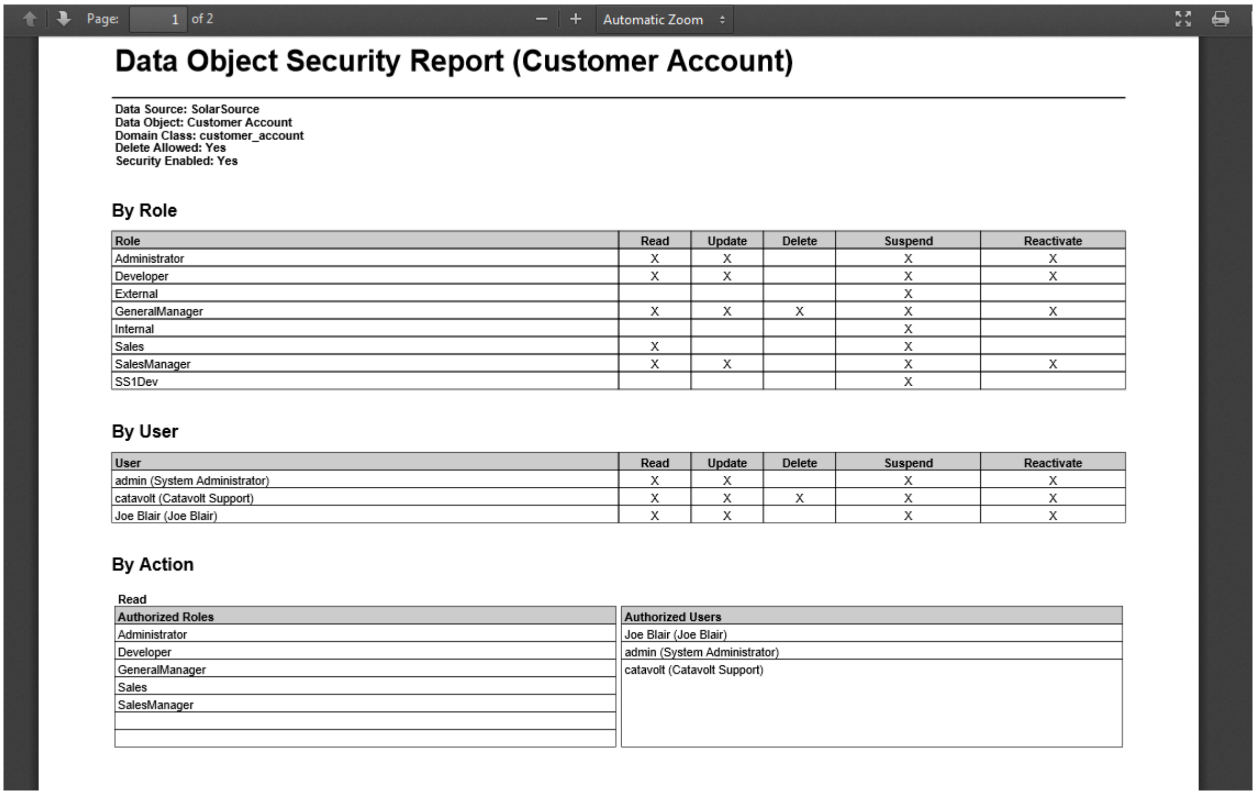
Object Mode specifies to display the Detail in read-only or update mode when displaying an object.

Key Query Clause specifies the criteria that is used to determine which record in the list should be used in the dashboard. Note that the query clause specified must return exactly 1 record. If 0 or more than 1 record is returned by the query clause, an error will be displayed when running the launcher. If the dashboard needs to show a different record for each user that runs it, you should use User Properties in the query clause to tailor it to each user. See Appendix B: Specifying Where Clauses for more information about specifying Additional Where Clause values.

Print Security Report

Selecting this menu option will generate a PDF document that describes in detail the authorizations granted for this Data Object. It will provide basic security information for the Data Object along with a list of all Security Roles in the system and their authorizations. In addition, a list of all Users is included along with their “effective” authorizations. Effective authorizations are determined by combining all Security Roles that a user is a member of along with the permissions for each Security Role. Note that permissions are inclusive. If a user is a member of at least 1 Security Role with authorization to a Data Object, the user will be authorized.

The Data Object Security Report will look similar to the following picture:



Data Object Security Report (Customer Account)

Data Source: SolarSource
Data Object: Customer Account
Domain Class: customer_account
Delete Allowed: Yes
Security Enabled: Yes

By Role

Role	Read	Update	Delete	Suspend	Reactivate
Administrator	X	X		X	X
Developer	X	X		X	X
External				X	
GeneralManager	X	X	X	X	X
Internal				X	
Sales	X			X	
SalesManager	X	X		X	X
SS1Dev				X	

By User

User	Read	Update	Delete	Suspend	Reactivate
admin (System Administrator)	X	X		X	X
catavolt (Catavolt Support)	X	X	X	X	X
Joe Blair (Joe Blair)	X	X		X	X

By Action

Read

Authorized Roles
Administrator
Developer
GeneralManager
Sales
SalesManager

Authorized Users
Joe Blair (Joe Blair)
admin (System Administrator)
catavolt (Catavolt Support)

Figure 64: A sample of a generated Data Object Security Report

See Chapter 10: Security for more information about object security in Xalt.

Launch Data Object

Launch Data Object is a menu option that allows you to directly launch a Data Object without first having to add the Data Object's launcher to a Workbench. This menu option is used by developers to launch and test changes to a Data Object without having to first deploy it on a Workbench.

Refresh Meta Data From Server

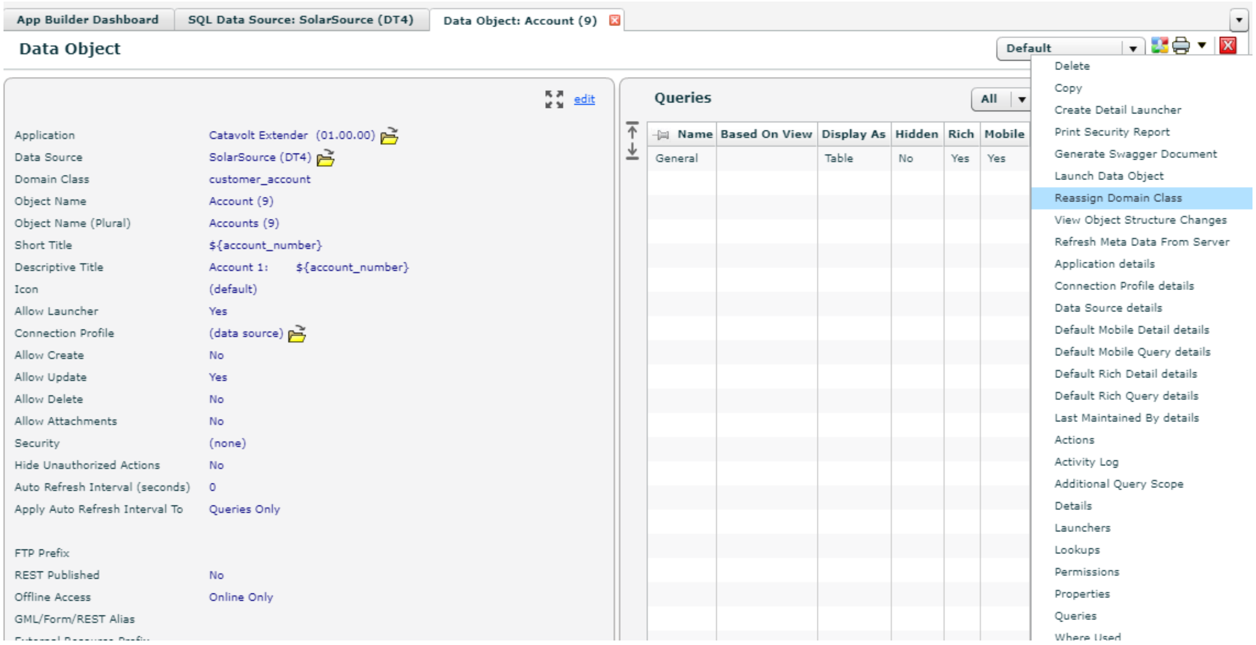
When you define Data Objects, Xalt retrieves metadata from the back-end system and stores it in the Xalt database for performance. In addition, once Xalt retrieves the metadata for a Data Object, it also caches it in memory for quick lookups. Normally, this is not a concern, since metadata in the back-end system tends to change rarely (if it all) once it has been defined. When the back-end system is changed (e.g. new objects are added, field lengths are changed, etc.), you need to notify Xalt to refresh its data and reset its caches. The Data Object's Refresh Meta Data From Server menu option will perform these tasks. It will go to the back-end system and re-fetch all metadata for the selected Data Object. In most cases, it will automatically make the necessary updates and deletes to the Data Object. In cases where Xalt cannot automatically make the changes (e.g. removing references to deleted fields in Additional Where Clause statements), Xalt will present you a list of references that you will need to manually update.

View Object Structure Changes

Some Xalt changes to metadata (e.g. defining key fields for a Data Object) are sent to the Connector gateway so that they can be used in the future when new Data Objects over the same database table are created. This menu option will return the object structure changes that currently exist on the Connector gateway. This menu option only needs to be performed when requested by Hexagon Support.

Reassign Domain Class

Selecting this menu option allows you to change the Domain Class for a Data Object. As this option can be destructive if not used properly, we have added it to the Details menu only. The Data Object List menu will not have this menu option.

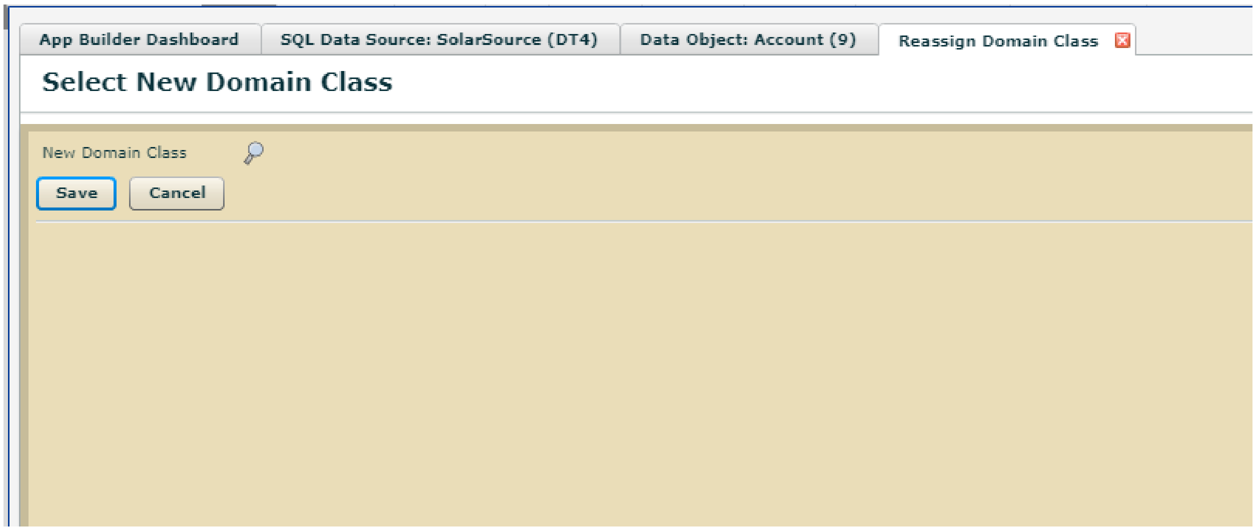


Application	Catavolt Extender (01.00.00)
Data Source	SolarSource (DT4)
Domain Class	customer_account
Object Name	Account (9)
Object Name (Plural)	Accounts (9)
Short Title	${account_number}
Descriptive Title	Account 1: ${account_number}
Icon	(default)
Allow Launcher	Yes
Connection Profile	(data source)
Allow Create	No
Allow Update	Yes
Allow Delete	No
Allow Attachments	No
Security	(none)
Hide Unauthorized Actions	No
Auto Refresh Interval (seconds)	0
Apply Auto Refresh Interval To	Queries Only
FTP Prefix	
REST Published	No
Offline Access	Online Only
GML/Form/REST Alias	
External Resource Prefix	

Name	Based On View	Display As	Hidden	Rich	Mobile
General		Table	No	Yes	Yes

Figure 65: The Data Object details view showing the Reassign Domain Class menu option

Selecting this menu action will bring up a prompt to choose a new Domain Class (similar to the Domain Class field when creating a new Data Object):



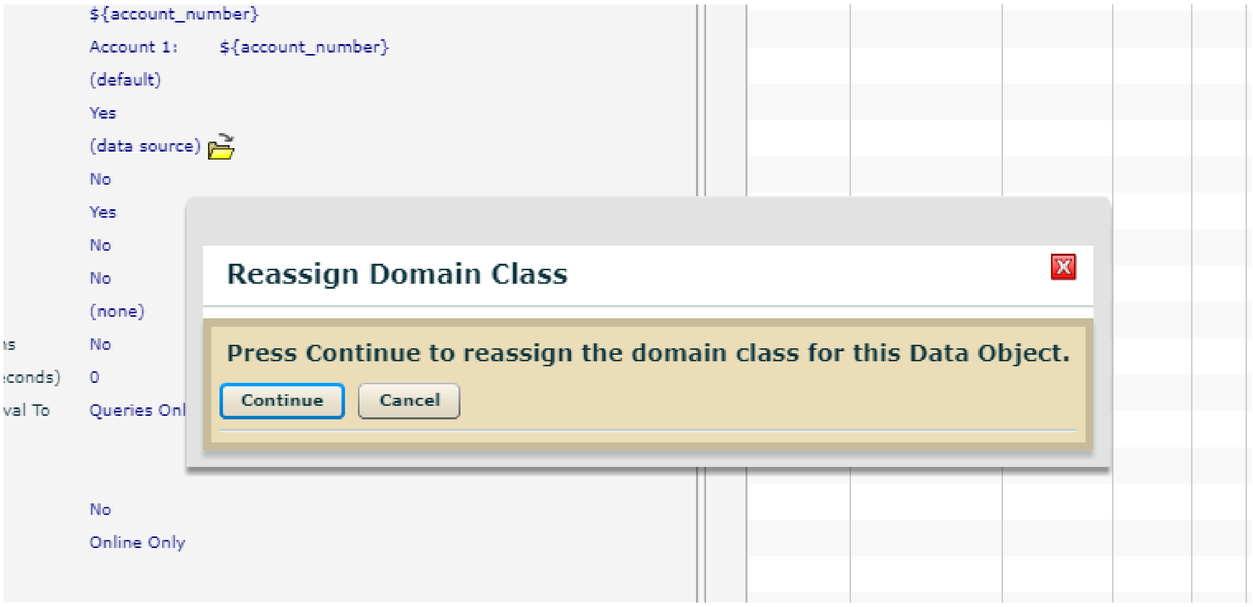
Select New Domain Class

New Domain Class

Figure 66: The Reassign Domain Class details view

When selecting a Domain Class, the server will run a number of checks to determine if any potential issues exist with renaming the Domain Class.

If no potential issues are found, a confirmation dialog is presented to complete domain class reassignment:

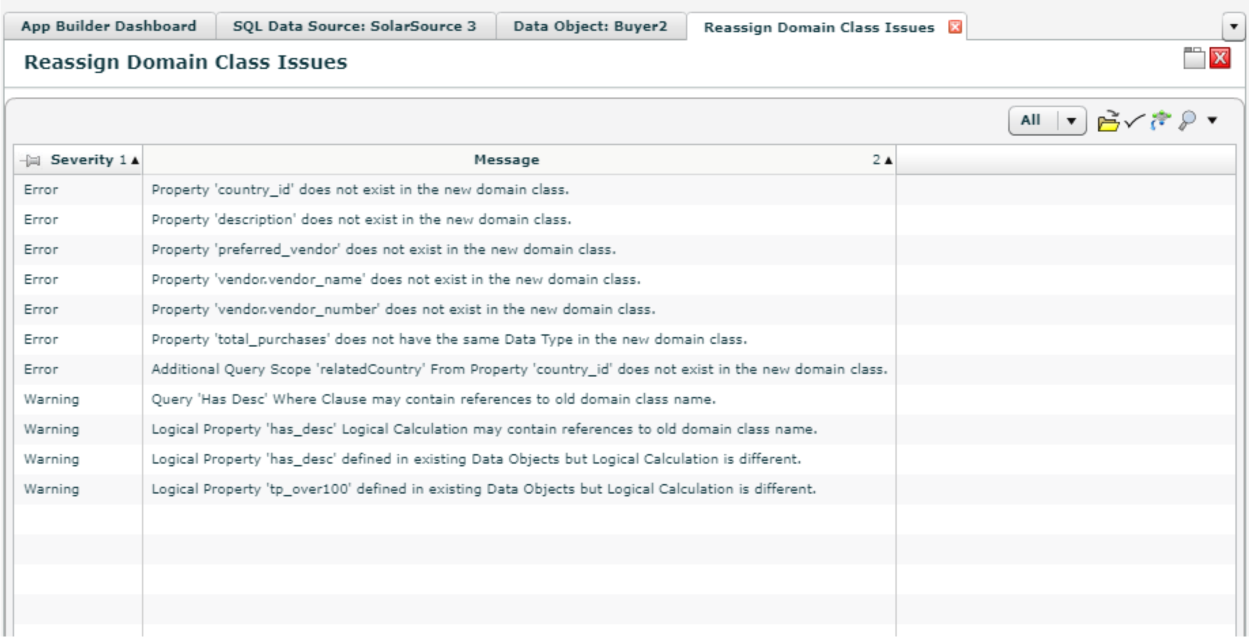


Reassign Domain Class

Press Continue to reassign the domain class for this Data Object.

Figure 67: The Reassign Domain Class confirmation prompt

If any potential issues are found, a list of the issues will be presented:



Reassign Domain Class Issues

Severity 1 ▲	Message 2 ▲
Error	Property 'country_id' does not exist in the new domain class.
Error	Property 'description' does not exist in the new domain class.
Error	Property 'preferred_vendor' does not exist in the new domain class.
Error	Property 'vendor.vendor_name' does not exist in the new domain class.
Error	Property 'vendor.vendor_number' does not exist in the new domain class.
Error	Property 'total_purchases' does not have the same Data Type in the new domain class.
Error	Additional Query Scope 'relatedCountry' From Property 'country_id' does not exist in the new domain class.
Warning	Query 'Has Desc' Where Clause may contain references to old domain class name.
Warning	Logical Property 'has_desc' Logical Calculation may contain references to old domain class name.
Warning	Logical Property 'has_desc' defined in existing Data Objects but Logical Calculation is different.
Warning	Logical Property 'tp_over100' defined in existing Data Objects but Logical Calculation is different.

Figure 68: The Reassign Domain Class Issues list view

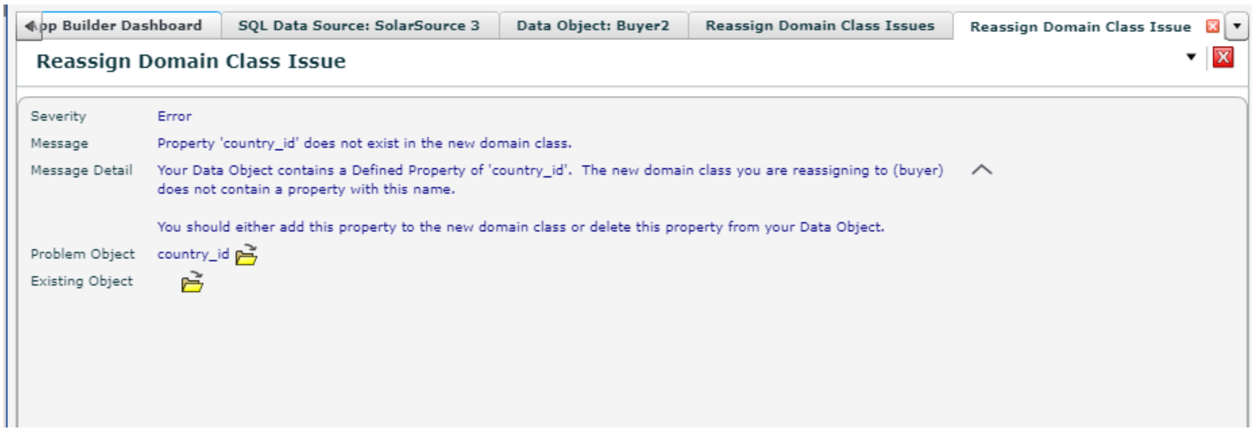
The list of potential issues will contain one or more of the following:

1. A Defined Property does not exist in the new domain class
2. An Additional Query Scope "From" Property does not exist in the new domain class
3. A Defined Property does not have the same Data Type in the new domain class
4. The current Data Object has a different key than the new domain class
5. A Where clause value references the old domain class name
6. A Logical calculation value references the old domain class name

Furthermore, if there are already other Data Objects defined over the new domain class you are choosing, the following issues may exist:

7. A logical field exists for the old domain class name but does not exist in the new domain class name
8. A logical field exists in both old and new domain class name but the logical calculation is different
9. The key properties defined in the old and new domain class names are different
10. An Additional Query Scope exists for the old domain class name but does not exist in the new domain class name
11. An Additional Query Scope exists in both old and new domain class names but is different

You can double-click any record to get more information about the issue:


App Builder Dashboard
SQL Data Source: SolarSource 3
Data Object: Buyer2
Reassign Domain Class Issues
Reassign Domain Class Issue

Reassign Domain Class Issue

Severity Error

Message Property 'country_id' does not exist in the new domain class.

Message Detail Your Data Object contains a Defined Property of 'country_id'. The new domain class you are reassigning to (buyer) does not contain a property with this name.

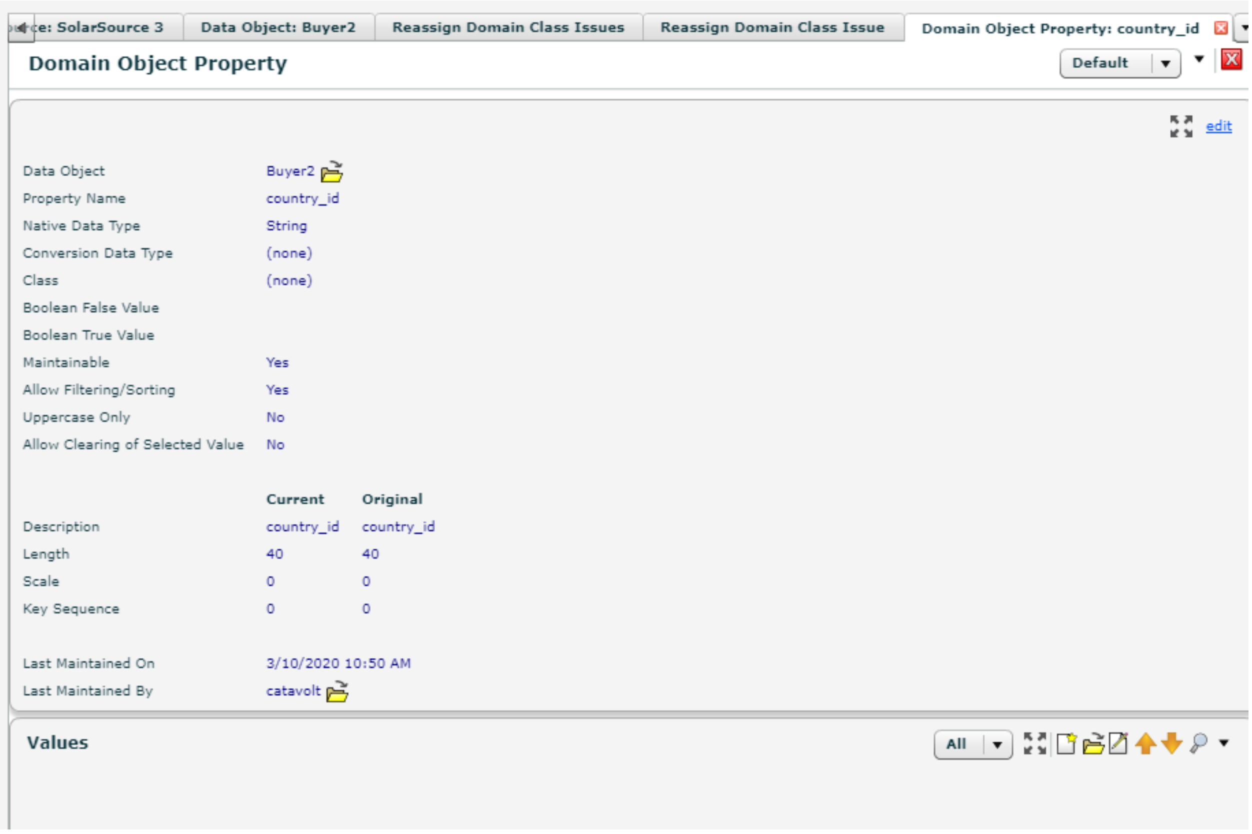
You should either add this property to the new domain class or delete this property from your Data Object.

Problem Object country_id

Existing Object

Figure 69: The Reassign Domain Class Issue details view

The Issue details will contain more detailed information about the potential issue and what should be done to resolve it. In addition, a direct link to the problem object is available. Clicking the open button next to it will bring up the object directly for review:


ce: SolarSource 3
Data Object: Buyer2
Reassign Domain Class Issues
Reassign Domain Class Issue
Domain Object Property: country_id
Default
edit
All

Domain Object Property

Data Object Buyer2
Property Name country_id
Native Data Type String
Conversion Data Type (none)
Class (none)
Boolean False Value
Boolean True Value
Maintainable Yes
Allow Filtering/Sorting Yes
Uppercase Only No
Allow Clearing of Selected Value No

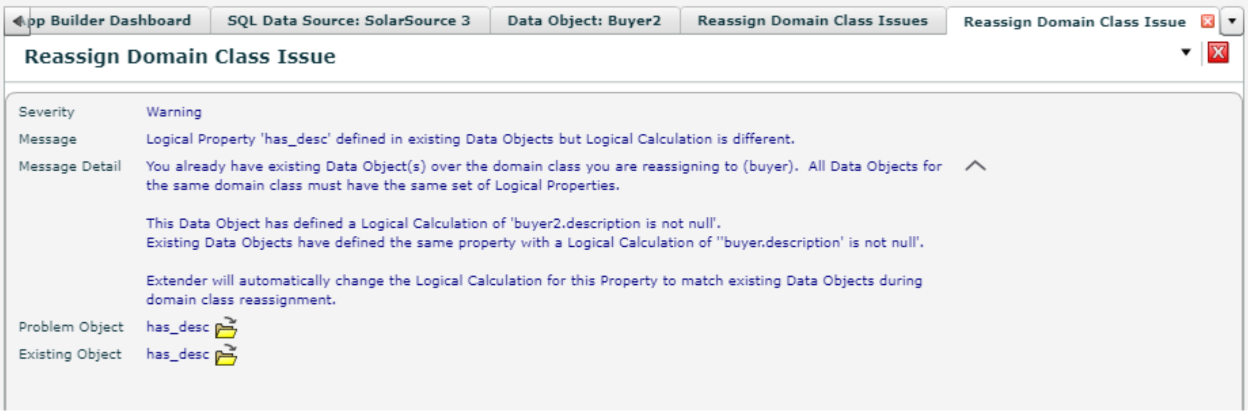
	Current	Original
Description	country_id	country_id
Length	40	40
Scale	0	0
Key Sequence	0	0

Last Maintained On 3/10/2020 10:50 AM
Last Maintained By catavolt

Values

Figure 70: The view that results from clicking the Problem Object open action

In some cases, you may already have Data Objects defined over the new domain class that contain information that conflicts with the Data Object you wish to reassign. In these cases, a direct link to the existing object is available:

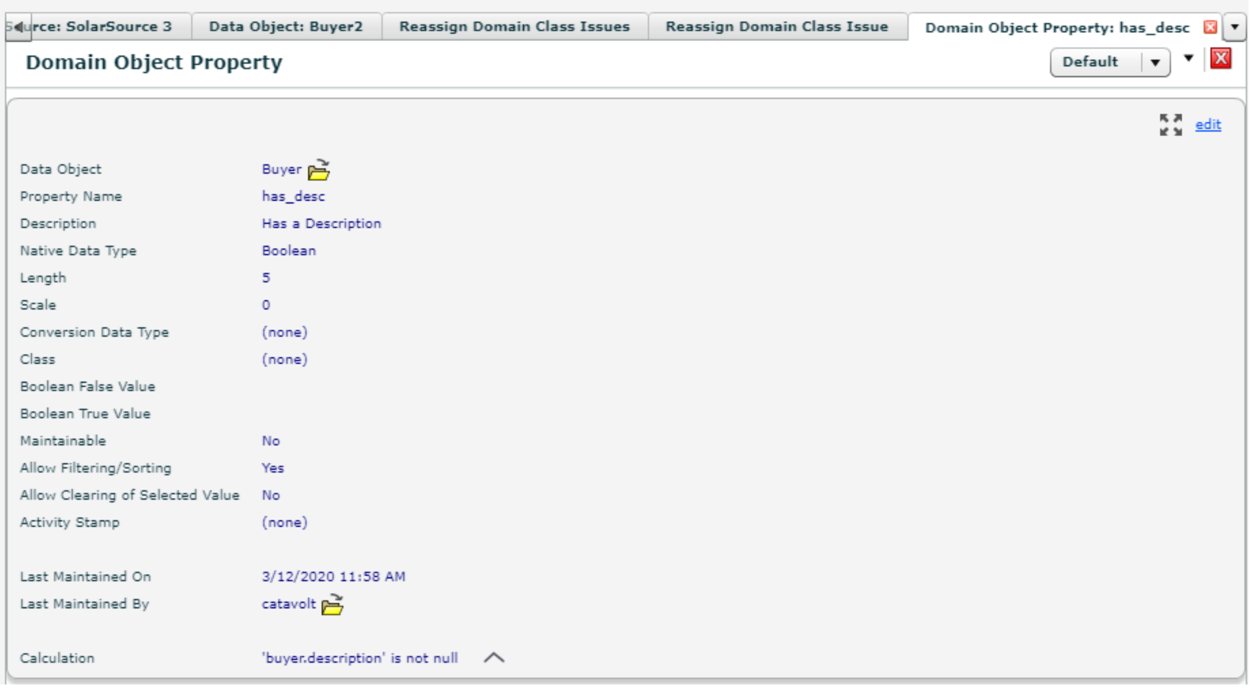

App Builder Dashboard
SQL Data Source: SolarSource 3
Data Object: Buyer2
Reassign Domain Class Issues
Reassign Domain Class Issue

Reassign Domain Class Issue

Severity	Warning
Message	Logical Property 'has_desc' defined in existing Data Objects but Logical Calculation is different.
Message Detail	You already have existing Data Object(s) over the domain class you are reassigning to (buyer). All Data Objects for the same domain class must have the same set of Logical Properties. This Data Object has defined a Logical Calculation of 'buyer2.description is not null'. Existing Data Objects have defined the same property with a Logical Calculation of ''buyer.description' is not null'. Extender will automatically change the Logical Calculation for this Property to match existing Data Objects during domain class reassignment.
Problem Object	has_desc
Existing Object	has_desc

Figure 71: The Reassign Domain Class Issue details view for a conflicting issue

Clicking the open button next to it will bring up the object directly for review:


Source: SolarSource 3
Data Object: Buyer2
Reassign Domain Class Issues
Reassign Domain Class Issue
Domain Object Property: has_desc
Default
edit

Domain Object Property

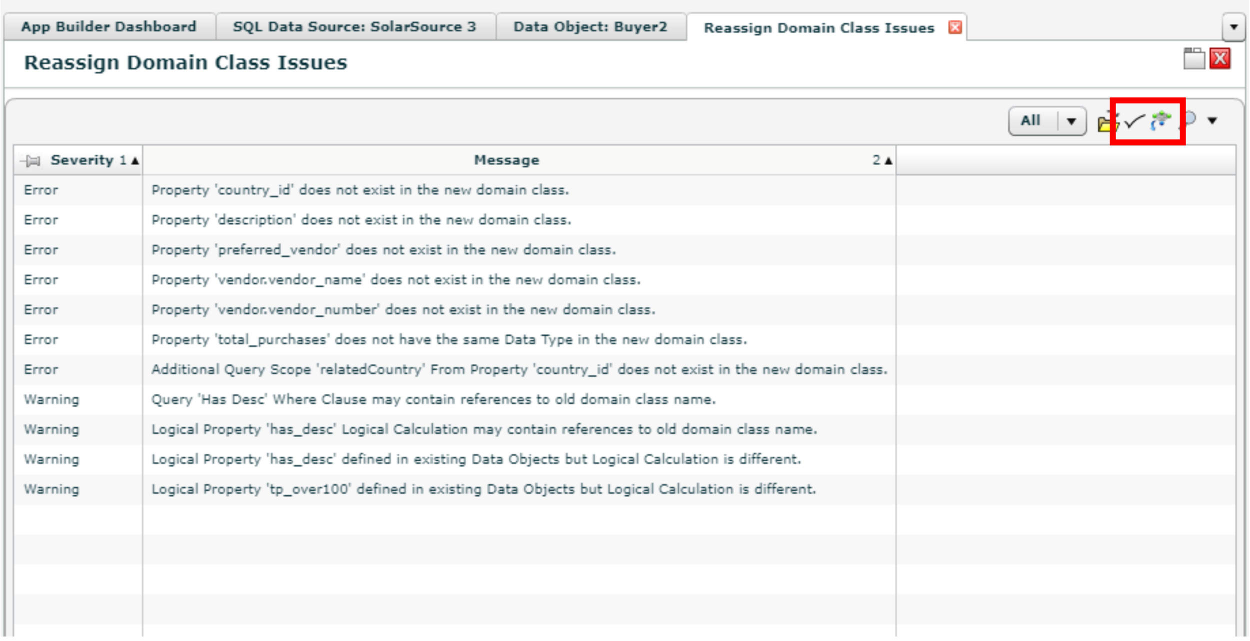
Data Object	Buyer
Property Name	has_desc
Description	Has a Description
Native Data Type	Boolean
Length	5
Scale	0
Conversion Data Type	(none)
Class	(none)
Boolean False Value	
Boolean True Value	
Maintainable	No
Allow Filtering/Sorting	Yes
Allow Clearing of Selected Value	No
Activity Stamp	(none)
Last Maintained On	3/12/2020 11:58 AM
Last Maintained By	catavolt
Calculation	'buyer.description' is not null

Figure 72: The view that results from clicking the Problem Object open action

The reported issues have 3 different Severities: Error, Warning, and Information:

- Error indicates a definite problem that the user will need to resolve in order to get the Data Object to work correctly after a reassignment.
- Warning indicates a potential problem that user should review to verify whether an actual problem exists, or indicates a problem that Xalt will automatically resolve in a way that may change the Data Object's behavior (e.g. changing the Data Object's key to match that of existing Data Objects over the same domain class).
- Information indicates a problem that Xalt will automatically resolve in a way that should not change the Data Object's behavior but the user should know about it anyway (e.g. adding a Logical Property from this Data Object to existing Data Objects over the same domain class).

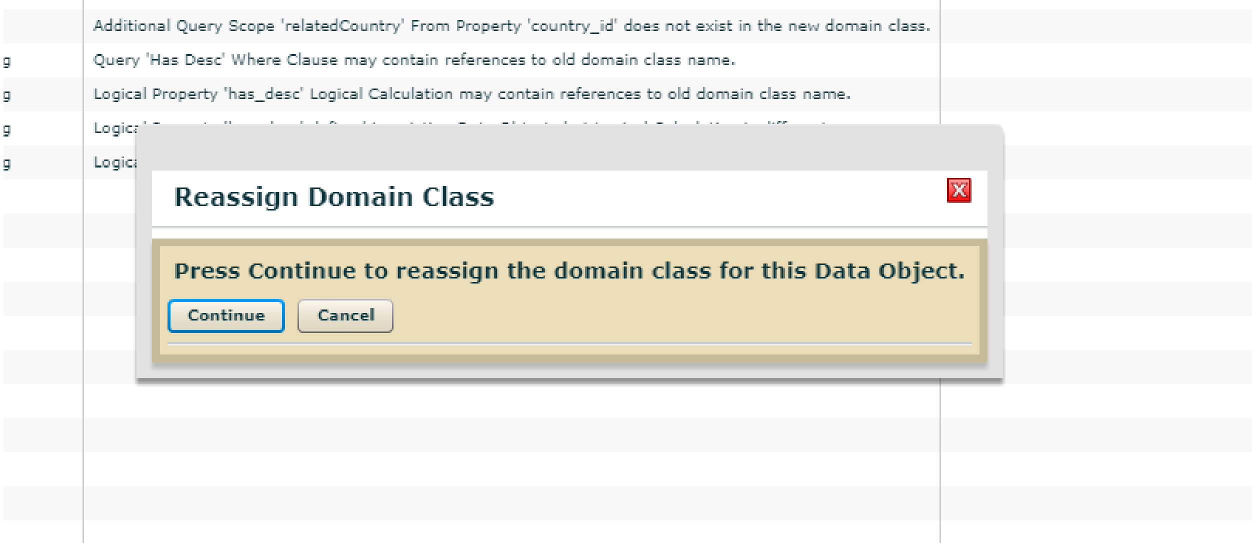
You can choose to resolve none, some, or all of the above issues before reassigning the domain class. The menu contains a "Recheck for Issues" action that will run the checks again and update the list of issues, removing any that have been resolved. There is also a "Confirm Reassign Domain Class" action that will immediately reassign the Domain Class regardless of whether issues remain:


App Builder Dashboard
SQL Data Source: SolarSource 3
Data Object: Buyer2
Reassign Domain Class Issues
Reassign Domain Class Issues
All

Severity 1 ▲	Message 2 ▲
Error	Property 'country_id' does not exist in the new domain class.
Error	Property 'description' does not exist in the new domain class.
Error	Property 'preferred_vendor' does not exist in the new domain class.
Error	Property 'vendor.vendor_name' does not exist in the new domain class.
Error	Property 'vendor.vendor_number' does not exist in the new domain class.
Error	Property 'total_purchases' does not have the same Data Type in the new domain class.
Error	Additional Query Scope 'relatedCountry' From Property 'country_id' does not exist in the new domain class.
Warning	Query 'Has Desc' Where Clause may contain references to old domain class name.
Warning	Logical Property 'has_desc' Logical Calculation may contain references to old domain class name.
Warning	Logical Property 'has_desc' defined in existing Data Objects but Logical Calculation is different.
Warning	Logical Property 'tp_over100' defined in existing Data Objects but Logical Calculation is different.

Figure 73: The Reassign Domain Class Issues list view highlighting the Recheck for Issues and Confirm Reassign Domain Class actions

Xalt allows you to go ahead and run the reassignment on demand, as some issues are recommended to be resolved AFTER domain class reassignment takes place. Choosing this menu option brings up the same confirmation prompt:



Additional Query Scope 'relatedCountry' From Property 'country_id' does not exist in the new domain class.

Query 'Has Desc' Where Clause may contain references to old domain class name.

Logical Property 'has_desc' Logical Calculation may contain references to old domain class name.

Reassign Domain Class

Press Continue to reassign the domain class for this Data Object.

Figure 74: The confirmation prompt that results from clicking the Confirm Reassign Domain Class action

After completing the reassignment, the issues list will remain open for you to investigate/resolve any remaining issues you did not resolve before the reassignment.

The domain class reassignment process will simply change the domain class name to match your choice.

If no other Data Objects in the system exist over the new domain class, the process is complete.

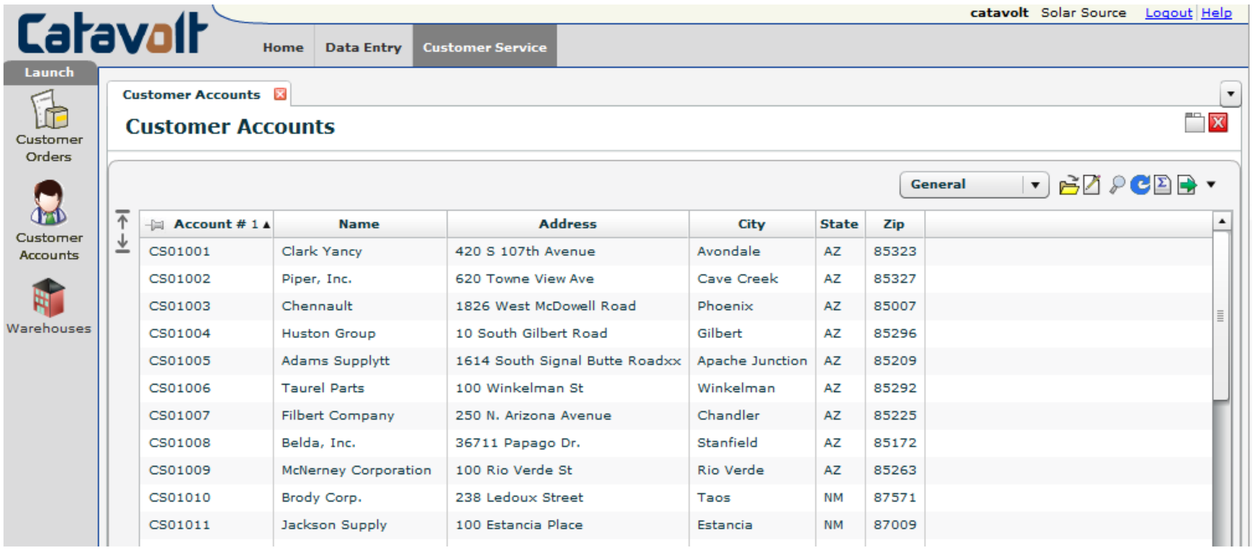
If existing Data Objects do exist over the new domain class name, a merge process will take place. This is because all Data Objects over the same domain class share the same Object Structure, which includes Logical Properties, Additional Query Scopes, and Key information. As part of the merge, any Logical Fields and Additional Query Scopes that exist in the “new” Data Object (the one being reassigned) but not in the “existing” Data Objects will be re-created in the “existing” Data Objects. In cases of items that exist in both but are different, the definition for the “new” Data Object (the one being reassigned) will be replaced with the corresponding definition from the “existing” Data Objects (reassigning a Data Object should force it to accept changes from other existing Data Object that are running in the tenant and not vice versa).

Chapter 4: Data Object Queries

Chapter Summary

A Query represents a list of records. At its core, a Query is made up of 3 parts: a list of what properties (fields, columns) to show, what order to show them in (sort), and (optionally) extra filtering logic to determine which subset of records to show. In Xalt, you are allowed to define multiple Queries for a particular Data Object. Each Query can provide a different view over the Data Object, so that you are not forced to put every possible field that may be of interest on a single Query.

For example, the General Query may show you address information about a Customer:

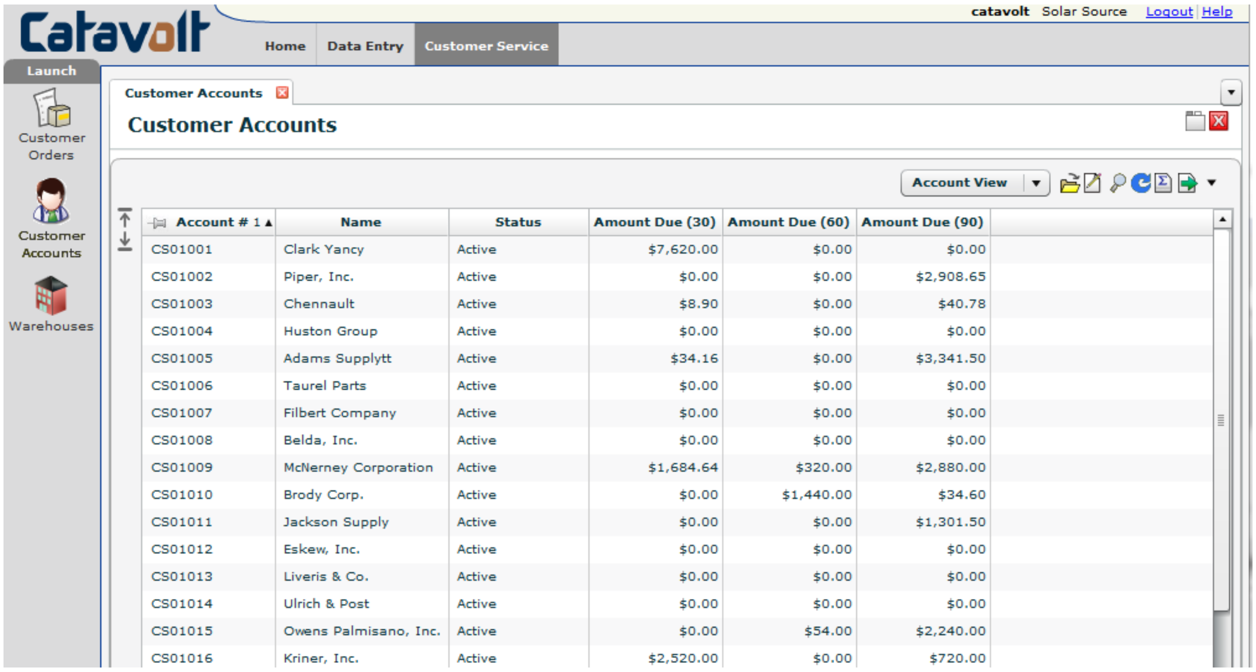


Customer Accounts

Account # 1 ▲	Name	Address	City	State	Zip
CS01001	Clark Yancy	420 S 107th Avenue	Avondale	AZ	85323
CS01002	Piper, Inc.	620 Towne View Ave	Cave Creek	AZ	85327
CS01003	Chennault	1826 West McDowell Road	Phoenix	AZ	85007
CS01004	Huston Group	10 South Gilbert Road	Gilbert	AZ	85296
CS01005	Adams Supplytt	1614 South Signal Butte Roadxx	Apache Junction	AZ	85209
CS01006	Taurel Parts	100 Winkelman St	Winkelman	AZ	85292
CS01007	Filbert Company	250 N. Arizona Avenue	Chandler	AZ	85225
CS01008	Belda, Inc.	36711 Papago Dr.	Stanfield	AZ	85172
CS01009	McNerney Corporation	100 Rio Verde St	Rio Verde	AZ	85263
CS01010	Brody Corp.	238 Ledoux Street	Taos	NM	87571
CS01011	Jackson Supply	100 Estancia Place	Estancia	NM	87009

Figure 1: An example Customer data object showing a query named General.

While the Account View query may only show account status information:



Customer Accounts

Account # 1 ▲	Name	Status	Amount Due (30)	Amount Due (60)	Amount Due (90)
CS01001	Clark Yancy	Active	$7,620.00	$0.00	$0.00
CS01002	Piper, Inc.	Active	$0.00	$0.00	$2,908.65
CS01003	Chennault	Active	$8.90	$0.00	$40.78
CS01004	Huston Group	Active	$0.00	$0.00	$0.00
CS01005	Adams Supplytt	Active	$34.16	$0.00	$3,341.50
CS01006	Taurel Parts	Active	$0.00	$0.00	$0.00
CS01007	Filbert Company	Active	$0.00	$0.00	$0.00
CS01008	Belda, Inc.	Active	$0.00	$0.00	$0.00
CS01009	McNerney Corporation	Active	$1,684.64	$320.00	$2,880.00
CS01010	Brody Corp.	Active	$0.00	$1,440.00	$34.60
CS01011	Jackson Supply	Active	$0.00	$0.00	$1,301.50
CS01012	Eskew, Inc.	Active	$0.00	$0.00	$0.00
CS01013	Liveris & Co.	Active	$0.00	$0.00	$0.00
CS01014	Ulrich & Post	Active	$0.00	$0.00	$0.00
CS01015	Owens Palmisano, Inc.	Active	$0.00	$54.00	$2,240.00
CS01016	Kriner, Inc.	Active	$2,520.00	$0.00	$720.00

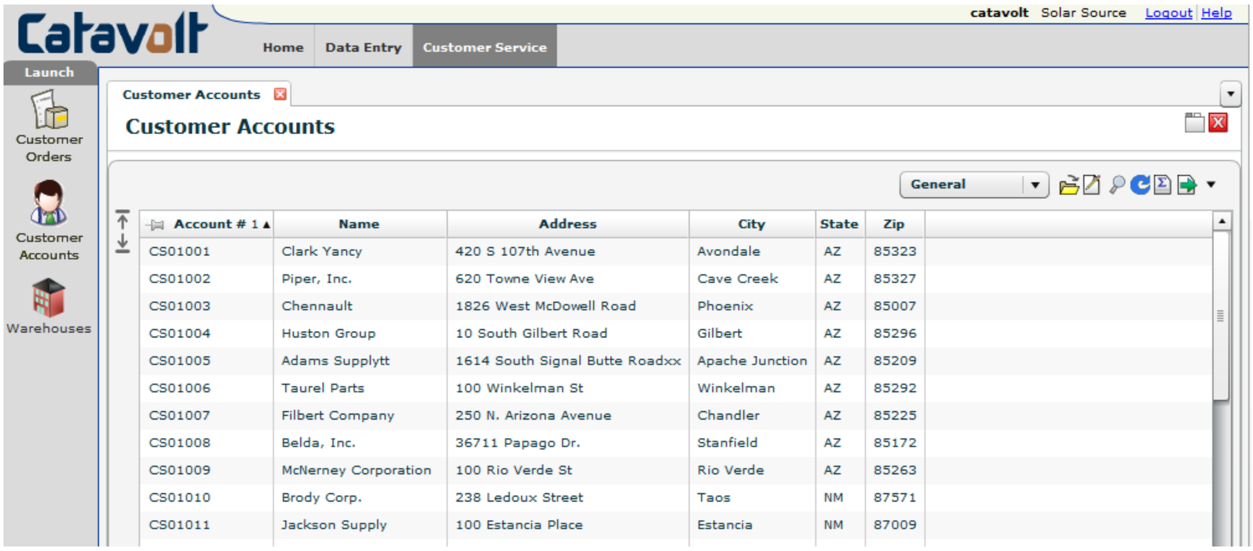
Figure 2: An example Customer data object showing a query named ‘Account View’

Types of Queries

When displaying a list of records, there are 5 main ways of rendering the data. These 5 types of Queries are Tables, Calendars, Graphs, Image Viewer, and Maps.

Table

A Table Query renders the records as a list. The list of records is displayed in a columnar table in the Web browser and Gadgets.



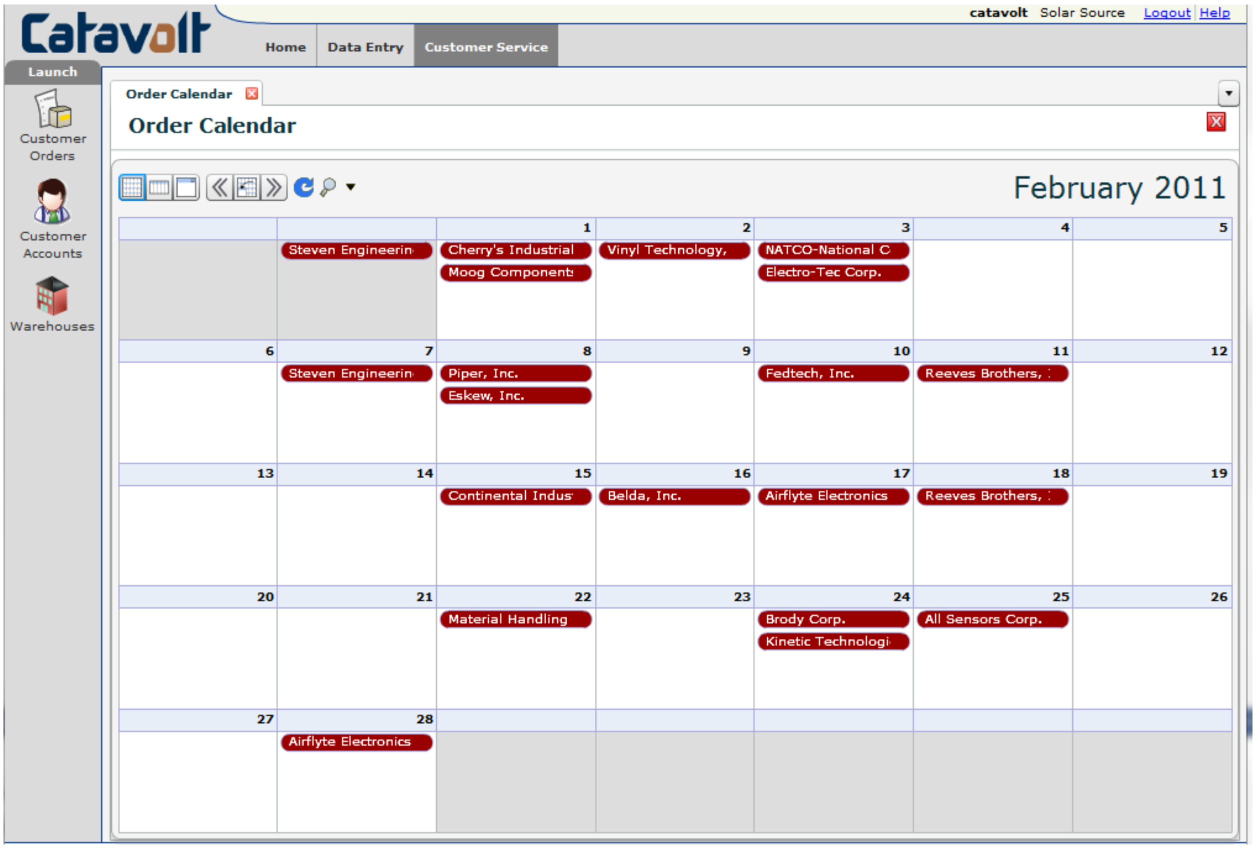
Customer Accounts

Account # 1 ▲	Name	Address	City	State	Zip
CS01001	Clark Yancy	420 S 107th Avenue	Avondale	AZ	85323
CS01002	Piper, Inc.	620 Towne View Ave	Cave Creek	AZ	85327
CS01003	Chennault	1826 West McDowell Road	Phoenix	AZ	85007
CS01004	Huston Group	10 South Gilbert Road	Gilbert	AZ	85296
CS01005	Adams Supplytt	1614 South Signal Butte Roadxx	Apache Junction	AZ	85209
CS01006	Taurel Parts	100 Winkelman St	Winkelman	AZ	85292
CS01007	Filbert Company	250 N. Arizona Avenue	Chandler	AZ	85225
CS01008	Belda, Inc.	36711 Papago Dr.	Stanfield	AZ	85172
CS01009	McNerney Corporation	100 Rio Verde St	Rio Verde	AZ	85263
CS01010	Brody Corp.	238 Ledoux Street	Taos	NM	87571
CS01011	Jackson Supply	100 Estancia Place	Estancia	NM	87009

Figure 3: Example Table query running in a web browser.

Calendar

A Calendar Query renders the records in a Calendar control by placing each record in the Calendar according to its Start Date/End Date/Occur Date properties.



	Steven Engineerin	1 Cherry's Industrial Moog Component:	2 Vinyl Technology,	3 NATCO-National C Electro-Tec Corp.	4	5
6	7 Steven Engineerin	8 Piper, Inc. Eskew, Inc.	9	10 Fedtech, Inc.	11 Reeves Brothers, :	12
13	14	15 Continental Indus	16 Belda, Inc.	17 Airflyte Electronics	18 Reeves Brothers, :	19
20	21	22 Material Handling	23	24 Brody Corp. Kinetic Technologi	25 All Sensors Corp.	26
27	28 Airflyte Electronics					

Figure 4: Example Calendar Query running in a web browser

Graph

A Graph Query renders the records using a graph. You can choose between a number of different graph types. In addition, you can create multi-level graphs by grouping similar records together, provide graph filters to allow real-time filtering of records, etc.

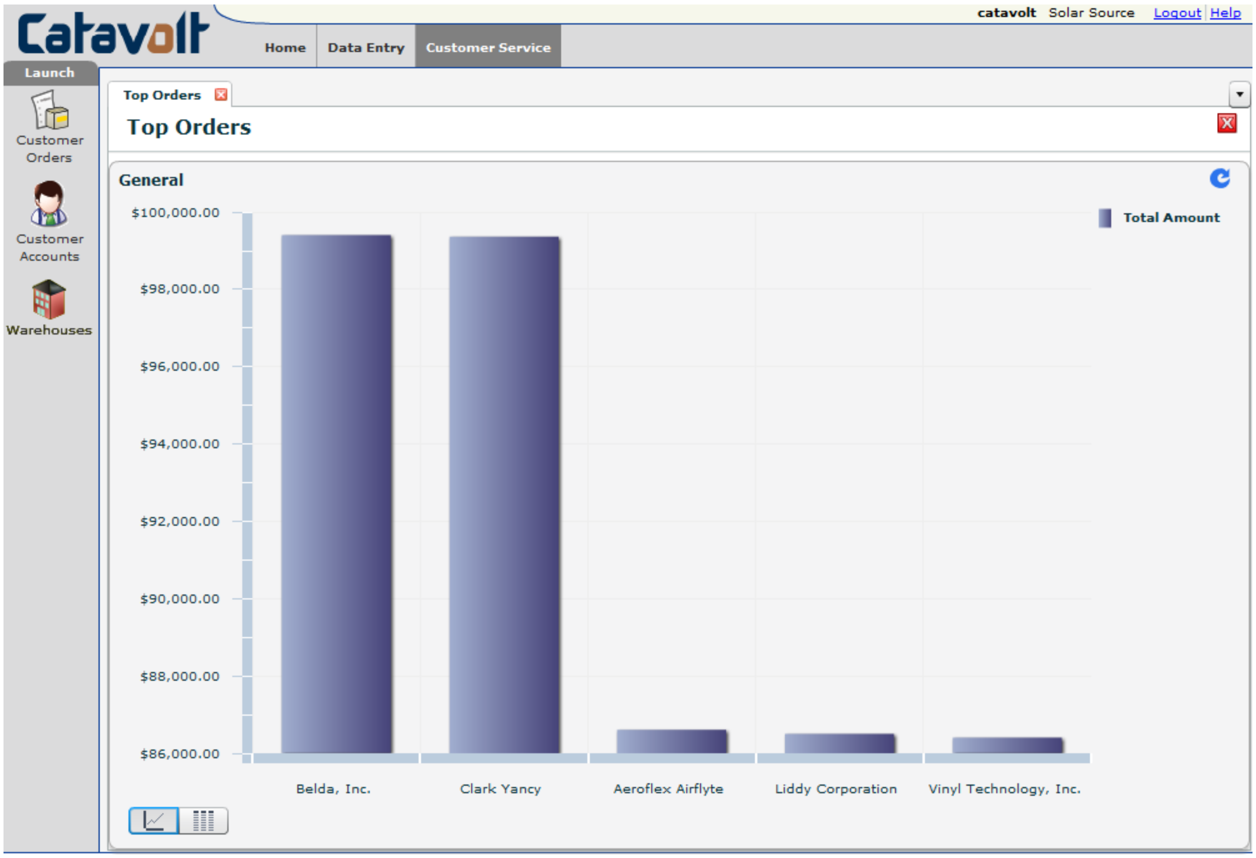
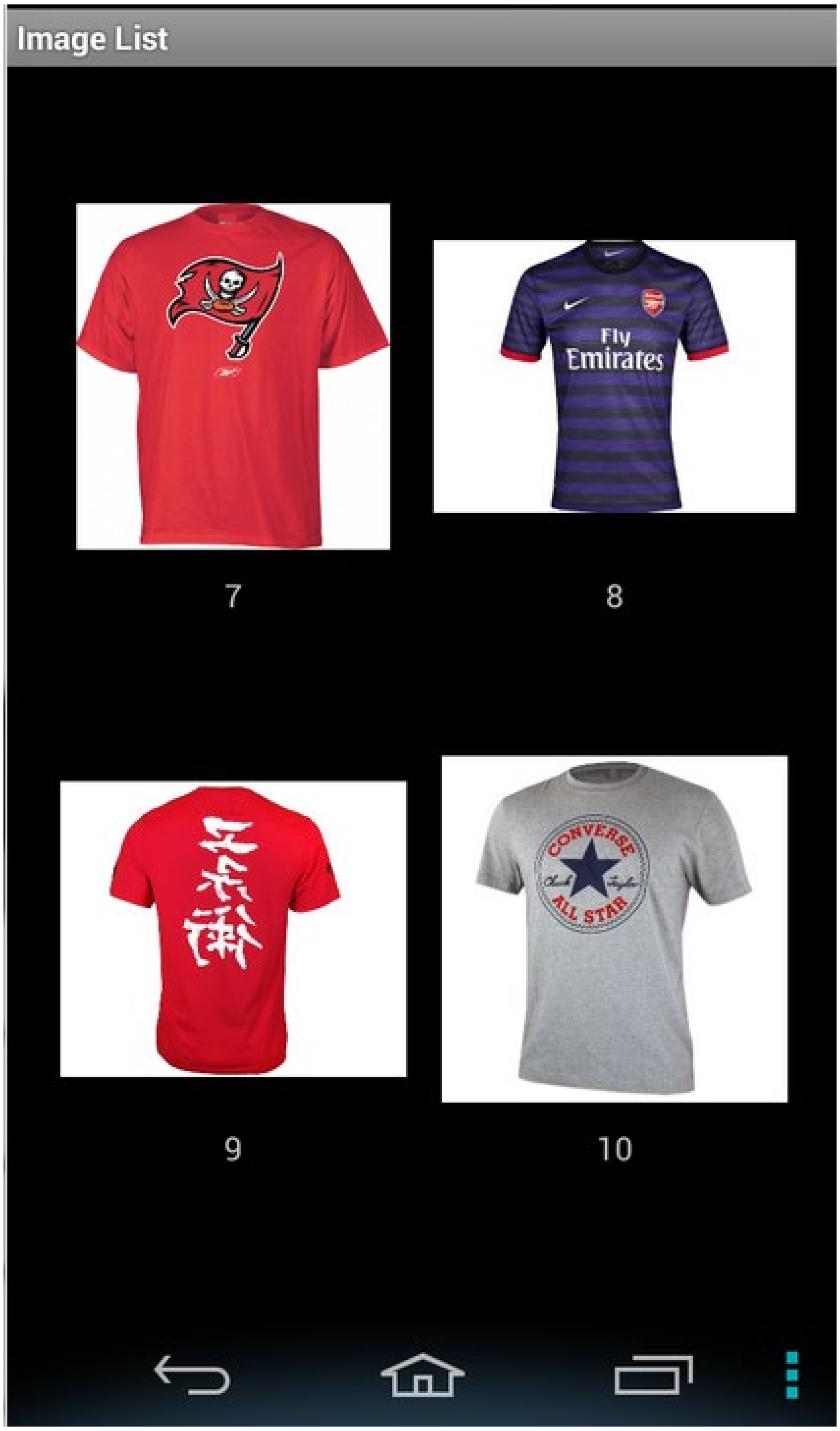


Figure 5: Example Graph Query running in a web browser

Image Viewer

An Image Viewer renders records as individual images that can be scrolled through en masse.



Map

A Map Query renders the records in a Map control by placing each record on the Map according to its Location properties.

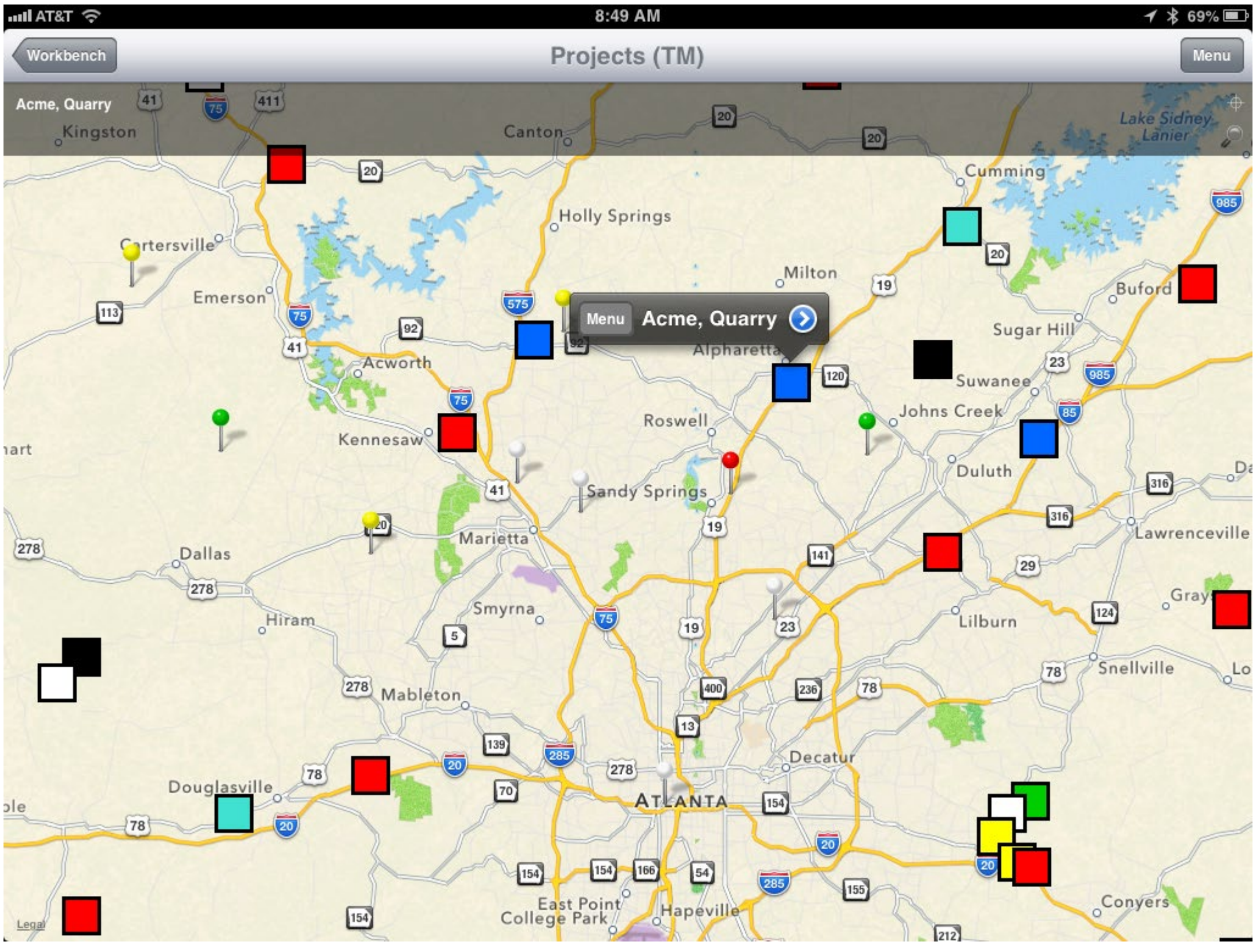
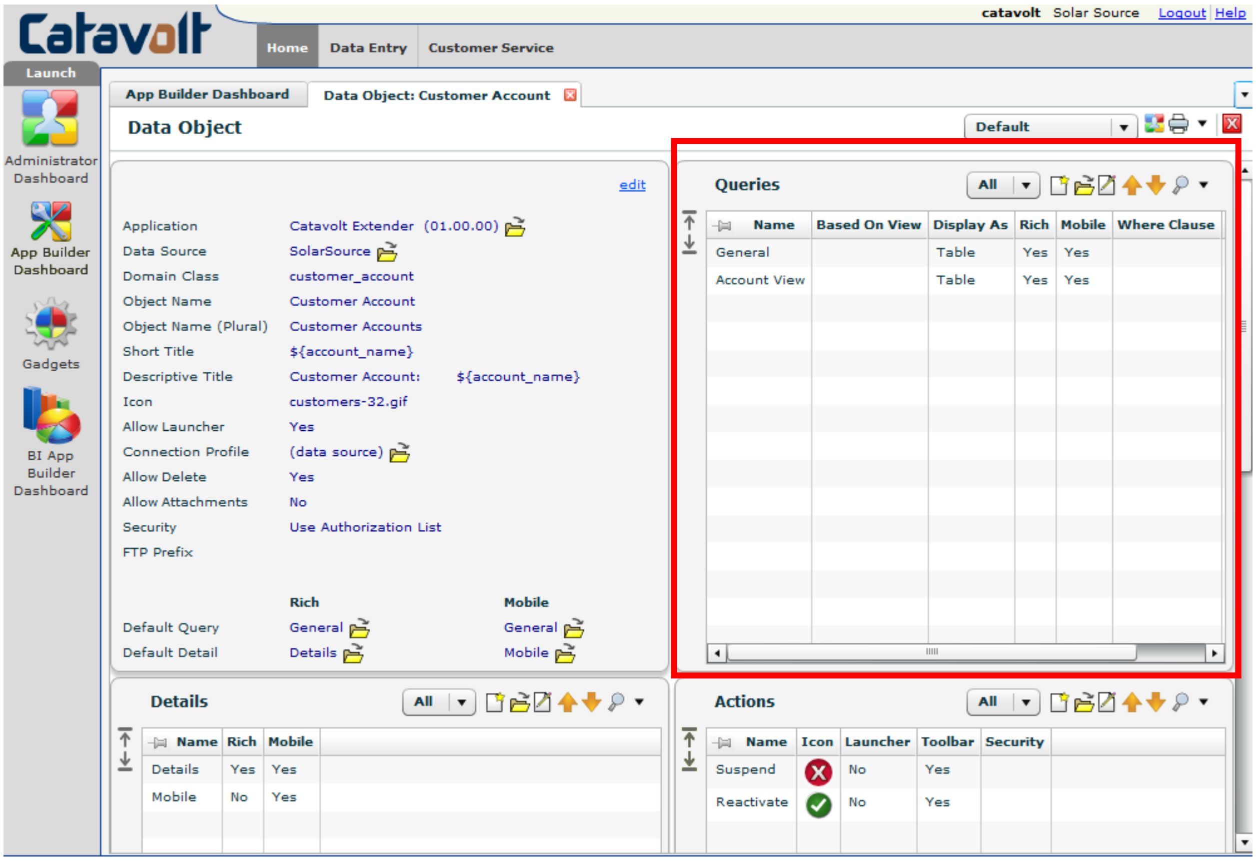


Figure 6: Example Map Query running on an iPad

Accessing Queries

To access the Queries defined for a particular Data Object, open the Data Object and look at the Default Detail. Multiple Queries can be created for a Data Object. They will appear in the Query dropdown list in the same order they appear below. To change the order, select a Query and press the Move Up and Move Down toolbar buttons.



Data Object

Application	Catavolt Extender (01.00.00)	
Data Source	SolarSource	
Domain Class	customer_account	
Object Name	Customer Account	
Object Name (Plural)	Customer Accounts	
Short Title	${account_name}	
Descriptive Title	Customer Account:	${account_name}
Icon	customers-32.gif	
Allow Launcher	Yes	
Connection Profile	(data source)	
Allow Delete	Yes	
Allow Attachments	No	
Security	Use Authorization List	
FTP Prefix		
	Rich	**Mobile**
Default Query	General	General
Default Detail	Details	Mobile

Queries

Name	Based On View	Display As	Rich	Mobile	Where Clause
General		Table	Yes	Yes	
Account View		Table	Yes	Yes	

Details

Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

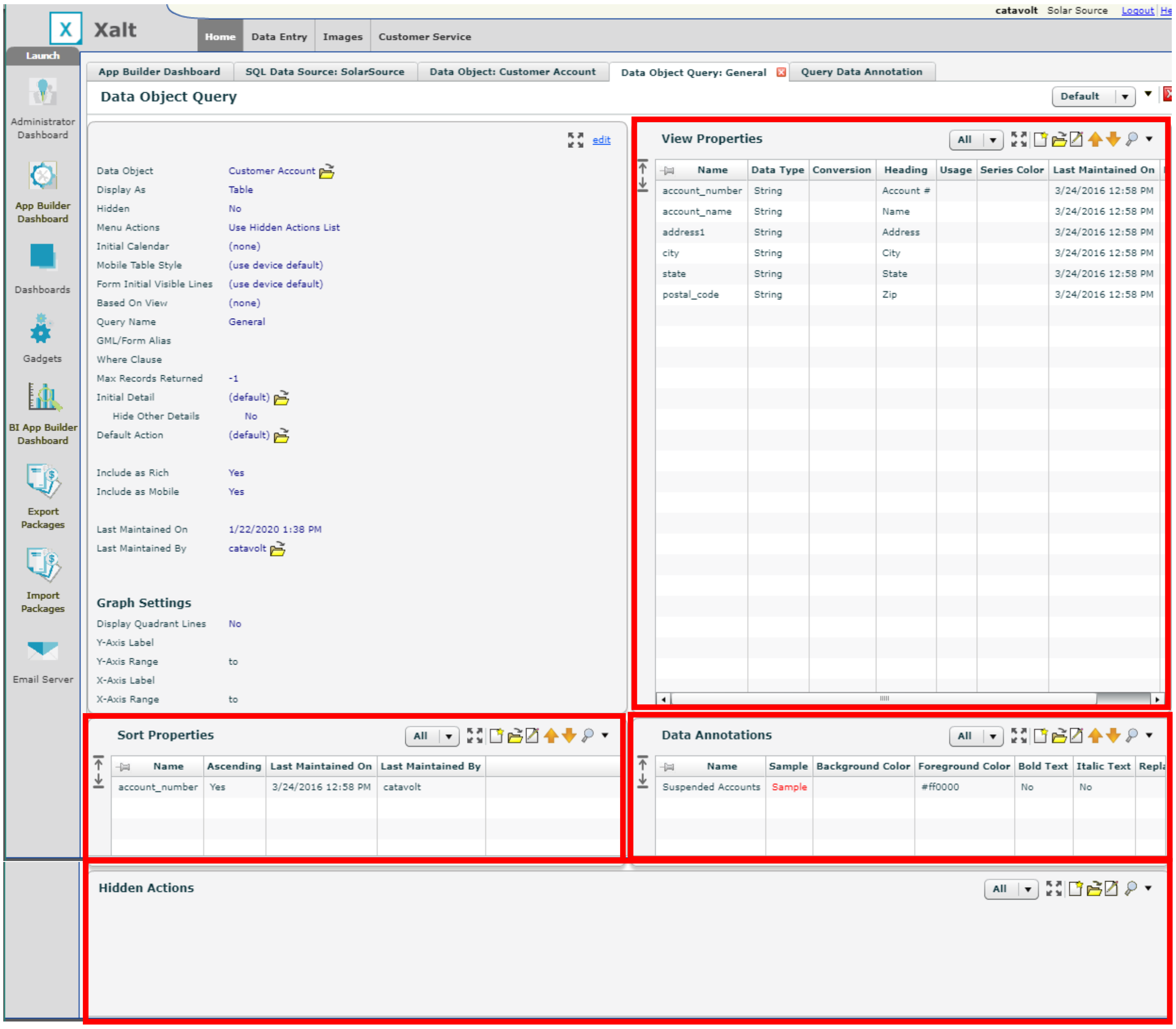
Actions

Name	Icon	Launcher	Toolbar	Security
Suspend	✗	No	Yes	
Reactivate	✓	No	Yes	

Figure 7: Data Object Query shown with the Default Details selected and the Queries query section highlighted

Query Components

When you display the definition for a Query, there are 2 Details that you can select from. The Default Detail shows you all the information about the Query. A Query is made contains 4 components: View Properties (a list of what properties to show), Sort Properties (what order to show them in), Data Annotations (the ability to annotate data using colors, fonts and text), and Hidden Actions (a list of Actions to hide from the menu when this Query is Active).



Data Object Query

Data Object	Customer Account
Display As	Table
Hidden	No
Menu Actions	Use Hidden Actions List
Initial Calendar	(none)
Mobile Table Style	(use device default)
Form Initial Visible Lines	(use device default)
Based On View	(none)
Query Name	General
GML/Form Alias	
Where Clause	
Max Records Returned	-1
Initial Detail	(default)
Hide Other Details	No
Default Action	(default)
Include as Rich	Yes
Include as Mobile	Yes
Last Maintained On	1/22/2020 1:38 PM
Last Maintained By	catavolt

Graph Settings

Display Quadrant Lines	No
Y-Axis Label	
Y-Axis Range	to
X-Axis Label	
X-Axis Range	to

View Properties

Name	Data Type	Conversion	Heading	Usage	Series Color	Last Maintained On
account_number	String		Account #			3/24/2016 12:58 PM
account_name	String		Name			3/24/2016 12:58 PM
address1	String		Address			3/24/2016 12:58 PM
city	String		City			3/24/2016 12:58 PM
state	String		State			3/24/2016 12:58 PM
postal_code	String		Zip			3/24/2016 12:58 PM

Sort Properties

Name	Ascending	Last Maintained On	Last Maintained By
account_number	Yes	3/24/2016 12:58 PM	catavolt

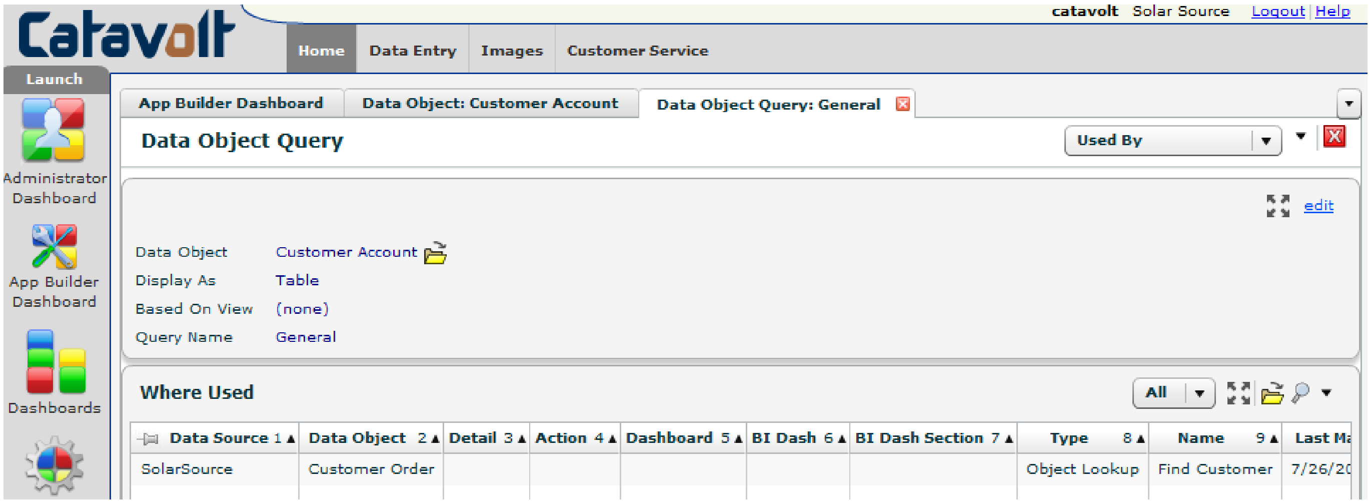
Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text
Suspended Accounts	Sample		#ff0000	No	No

Hidden Actions

Figure 8: Data Object Query details highlighting the query sections for View Properties, Sort Properties, Data Annotations and Hidden Actions

The Used By Detail shows a Where Used list of objects that are currently using this Query.



Data Object Query

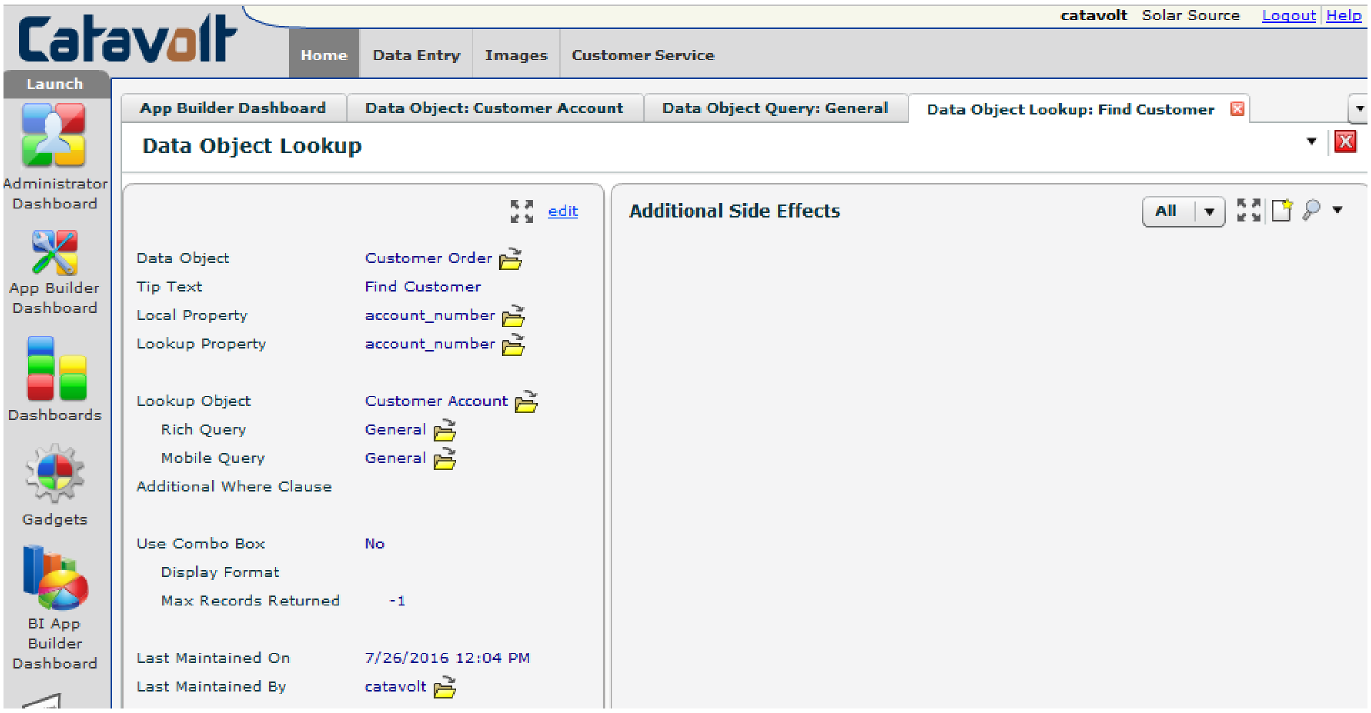
Data Object Customer Account
Display As Table
Based On View (none)
Query Name General

Where Used

Data Source 1	Data Object 2	Detail 3	Action 4	Dashboard 5	BI Dash 6	BI Dash Section 7	Type 8	Name 9	Last Ma
SolarSource	Customer Order						Object Lookup	Find Customer	7/26/20

Figure 9: Query 'Used By' details

Opening a Where Used record will take you directly to the object using the Query:



Data Object Lookup

Data Object Customer Order
Tip Text Find Customer
Local Property account_number
Lookup Property account_number

Lookup Object Customer Account
Rich Query General
Mobile Query General
Additional Where Clause

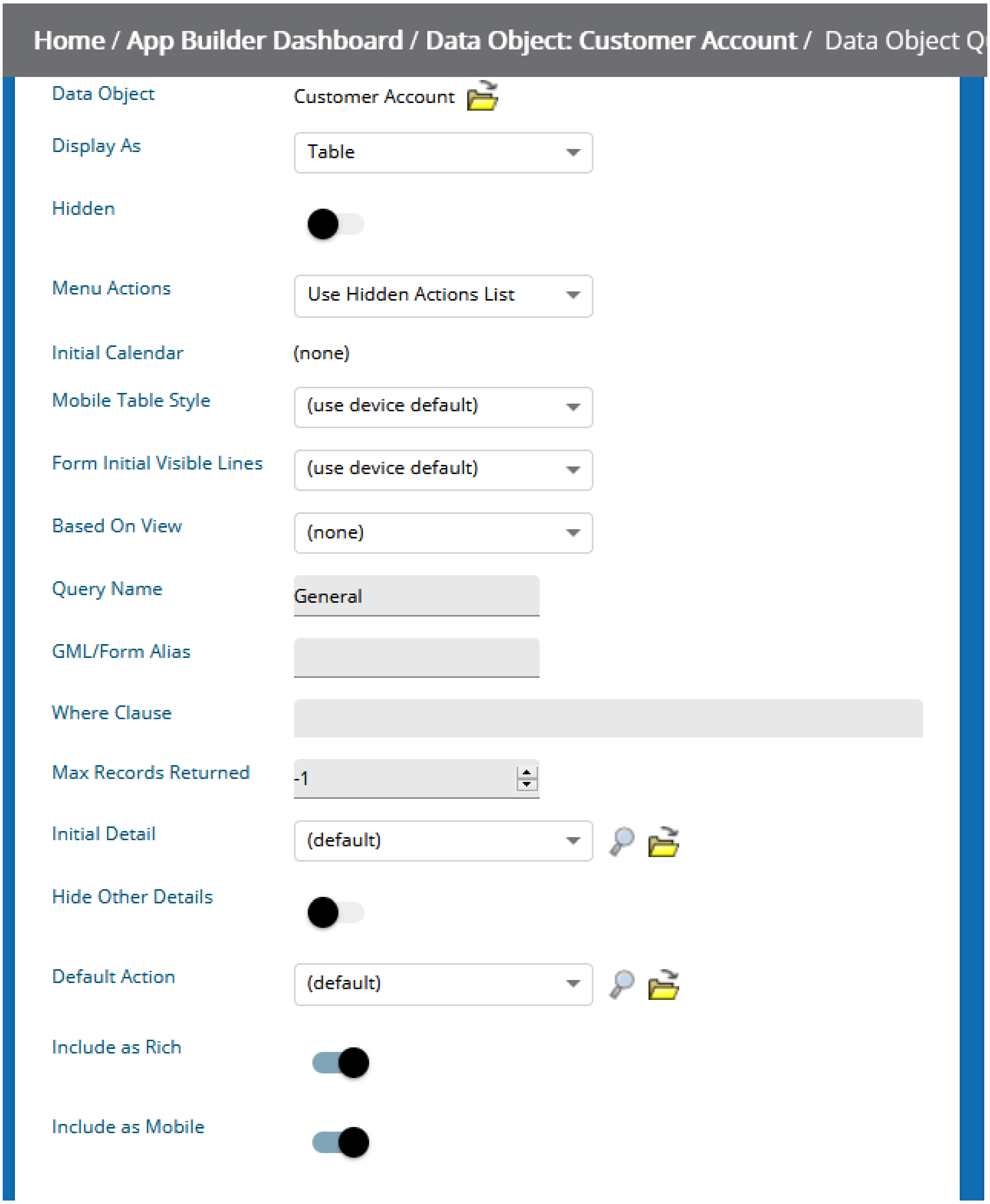
Use Combo Box No
Display Format
Max Records Returned -1

Last Maintained On 7/26/2016 12:04 PM
Last Maintained By catavolt

Additional Side Effects

Figure 10: Query Where Used

Creating Queries



Home / App Builder Dashboard / Data Object: Customer Account / Data Object Q

Data Object	Customer Account
Display As	Table
Hidden	○
Menu Actions	Use Hidden Actions List
Initial Calendar	(none)
Mobile Table Style	(use device default)
Form Initial Visible Lines	(use device default)
Based On View	(none)
Query Name	General
GML/Form Alias	
Where Clause	
Max Records Returned	-1
Initial Detail	(default)
Hide Other Details	○
Default Action	(default)
Include as Rich	●
Include as Mobile	●

Figure 11: The create Data Object Query view

When creating a Data Object Query, the first thing to decide is the Display As option. Display As specifies how to render the records in the Query. The available options are Bar Graph, Bubble Chart, Calendar, Donut Chart, Gantt Chart, Image Viewer, Line Graph, Map, Pie Graph, Scatter Plot, Stacked Bar Graph, and Table.

Hidden specifies whether this Query will appear in the runtime Query dropdown list. If Hidden is selected, this Query will not appear, nor will it appear in the Data Object's Default Rich/Mobile Query list. The Query can still be explicitly selected in other places (e.g. in a Detail Query Section or Completion Action Rich/Mobile Query).

Menu Actions – specifies how to display the menu when this Query is active. The list of available values are:

- (show all) – Show all Actions on the menu. This is the default
- (hide all) – Hide all Actions (standard as well as custom) from the menu. You can choose this option to quickly hide the entire menu without having to add every action to the Hidden Actions List.
- Read-only Standard Actions -- Hide all custom Actions as well any standard Actions that maintain data (Create, Update, Delete, and Attach File (for Infor XA System-Link Data Objects)). You can choose this option to quickly make the menu "read-only" without having to add every action to the Hidden Actions List.
- Use Hidden Actions List – Specify a list of Actions to hide from the menu when this Query is active.

Note that you can still call an action explicitly in a GML screen even if it is hidden from the menu.

Initial Calendar – specifies the initial view that comes up for Calendars. The list of available values are:

- Day View – Display one day per screen
- Week View – Display one week per screen
- Month View – Display on month per screen

Mobile Table Style – specifies the table style to use when displaying this query on a mobile device. The list of available values are:

- (use device default) – Allow the device to determine which format to use. Most tablets will default to Grid, and most phones will default to Form when the phone is upright (portrait) and Grid when the phone is sideways (landscape).
- Form – Force the query to display in Form style (See Figure 10: An example of a mobile Query in Form style).
- Grid – Force the query to display in Grid style (See Figure 11: An example of a mobile Query in Grid style).

Form Initial Visible Lines – specifies the number of properties that will initially display for each record when displaying this query on a mobile device. The user can see more or fewer properties once the query is displayed by using the 2 finger pinch gesture.

- (use device default) – Allow the device to determine the initial number of properties to display.
- (show all properties) – Display all properties in the query initially.
- 1-10 – Display from 1 to 10 properties initially.

Based On View specifies which predefined view you would like to use when creating a Query. Note that this option is only available for back-end systems (such as IDF System-Link) that have predefined views for Data Objects. The (none) option will be the only available option if your back-end system does not support this feature, or if you choose to create a Query using your own unique set of properties.

Query Name is used to uniquely identify this Query. It is presented to the user in the Query dropdown list.

GML/Form Alias allows you to specify an alias when using this Query with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Where Clause specifies criteria that can be used to filter the list of records to be displayed. If you do not specify a Where Clause, you will get a list of all records of the selected Data Object. Specifying a Where Clause allows you to do things like only show Orders that are active or whose order amounts exceed a certain threshold. See Appendix B: Specifying Where Clauses for more information about specifying Additional Where Clause values.

Max Records Returned specifies the number of records to return in the list. Xalt defaults this value to -1, which will return all records. Xalt loads data in small batches when displaying lists. For example, a Calendar query only pulls in records for the months that it has displayed. This allows Xalt to process back end tables that contain large amounts of data without negative performance impacts. There are however some situations that cannot use the small batch queries. The most notable of which is a graph in which all records need to be displayed to render an accurate graph. In these cases Xalt will limit the query to a maximum of 1000 records to prevent performance issues. Please see details in the Query Graph section of this manual for more details.

Initial Detail specifies which Data Object Detail will be displayed when you Open or Edit a record while using this Query. Xalt defaults this value to (default), which specifies to use the Default Rich or Mobile Detail as defined on the Data Object.

Hide Other Details specifies whether the runtime Details dropdown list will contain just the Initial Detail or all non-hidden Details. If Hide Other Details is selected, only the Initial Detail will be available to choose in the Details dropdown list.

Default Action specifies which action will be performed when you double-click a record (or perform the mobile-device equivalent of a double-click) while using this Query. All Actions for this Data Object will be included. In addition, the following choices are also included if their corresponding menu actions are available for the Data Object:

- (default) – Perform the Open action if it is available on the Data Object, otherwise perform the first Data Object Action in the list for the Data Object
- (none) – Do nothing when a double-click is initiated
- Delete – Perform the Delete action (deletes the selected record)
- Edit – Perform the Edit action (opens details in edit mode)
- Open – Perform the Open action (opens details in read mode)

Include as Rich specifies that this query is available to use on rich clients, such as the Web and Gadgets.

Include as Mobile specifies that this query is available to use on mobile clients.

The Maintainable Query Settings section contains values that are used when this Query is used as part of a Maintainable Query Dashboard. See Chapter 14: Dashboards for more information about creating a Maintainable Query Dashboard:

Hide Heading Groups specifies whether to hide or show any Heading Groups defined for Query View Properties associated with this Query. The default is No (show all Heading Groups). See the View Properties

The Graph Settings section contains values that are used when the Display As property is set to a Graph type:

Gantt Chart Hover Detail specifies a Detail to display as tip text when hovering over a bar in a Gantt Chart. Extender will pull the first property section from this detail to provide the labels and properties to use in the Hover Detail. Constants and blank lines are allowed (column breaks are ignored). If a Hover Detail is not specified, no tip text will be shown when hovering over an Item bar.

Display Quadrant Lines specifies whether to overlay quadrant lines on top of the graph. This option is available for Bubble Charts and Scatter Plots.

Y-Axis Label specifies a label to use for the Y-Axis of your graph. This option is available for all graphs except for Pie Graph and Donut Chart.

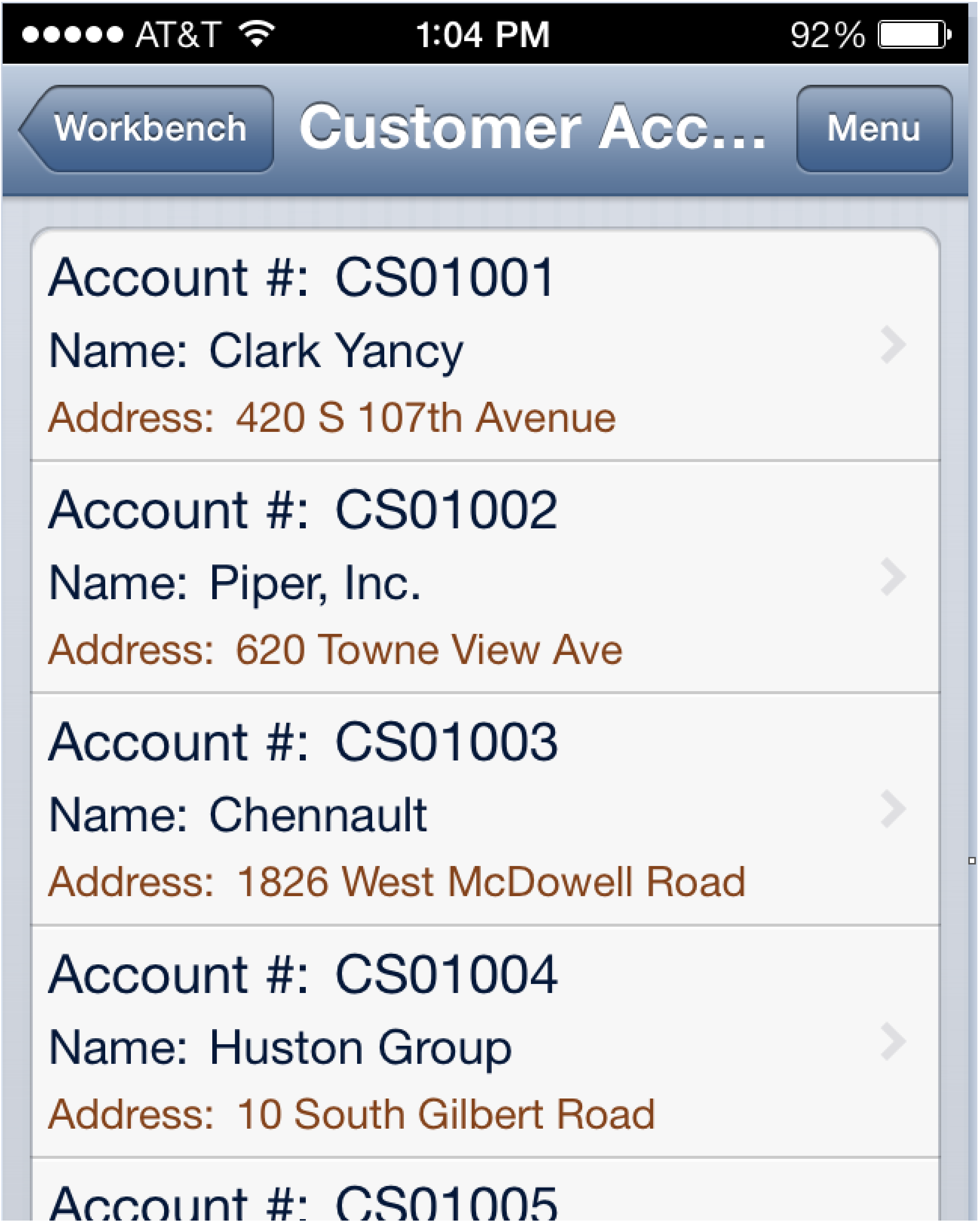
Y-Axis Range allows you to override part or all of graph's range for its Y-Axis. If these values are blank, the graph will auto-size the range to fit the graph's data. Note that you can specify one value while leaving the other blank (e.g. setting the From Range to 0 while leaving the To Range blank will anchor the bottom of the graph to 0 while letting the top of the graph float to fit the data. This option is available for all graphs except for Pie Graph and Donut Chart. Specifying a Y-Axis Range is required if you choose the Display Quadrant Lines option.

X-Axis Label specifies a label to use for the X-Axis of your graph. This option is available for all graphs except for Pie Graph and Donut Chart.

X-Axis Range allows you to override part or all of graph's range for its X-Axis. If these values are blank, the graph will auto-size the range to fit the graph's data. Note that you can specify one value while leaving the other blank (e.g. setting the From Range to 0 while leaving the To Range blank will anchor the left side of the graph to 0 while letting the right side of the graph float to fit the data. If the graph type is Bubble Chart or Scatter Plot, this option is available and will accept a numeric value. If the graph type is Gantt Chart, this option is available and will accept a Date or Timestamp value. You can use any of the following formats to enter a constant date/time:

- Dates: **YYYY-MM-DD** (2021-03-24) / **YYYYMMDD** (20210324) / **CYYMMDD** (2210324)
- Times: **HH:MM:SS** (12:10:58) / **HH:MM** (12:10) / **HHMMSS** (121058) / **HHMM** (1210)
- Timestamps: **DateTTime** (2021-03-24T12:10:58) / **Date Time** (2021-03-24 12:10:58)
- Standard Substitution values can also be used to specify floating dates, for example (${CURRENT_DATE-1WEEK@DATE}T${CURRENT_TIME+8HOURS@TIME})
- From Time defaults to 00:00:00 if not specified. To Time defaults to 23:59:59 if not specified.

Specifying an X-Axis Range is required if you choose the Display Quadrant Lines option.



Account #: CS01001
Name: Clark Yancy
Address: 420 S 107th Avenue

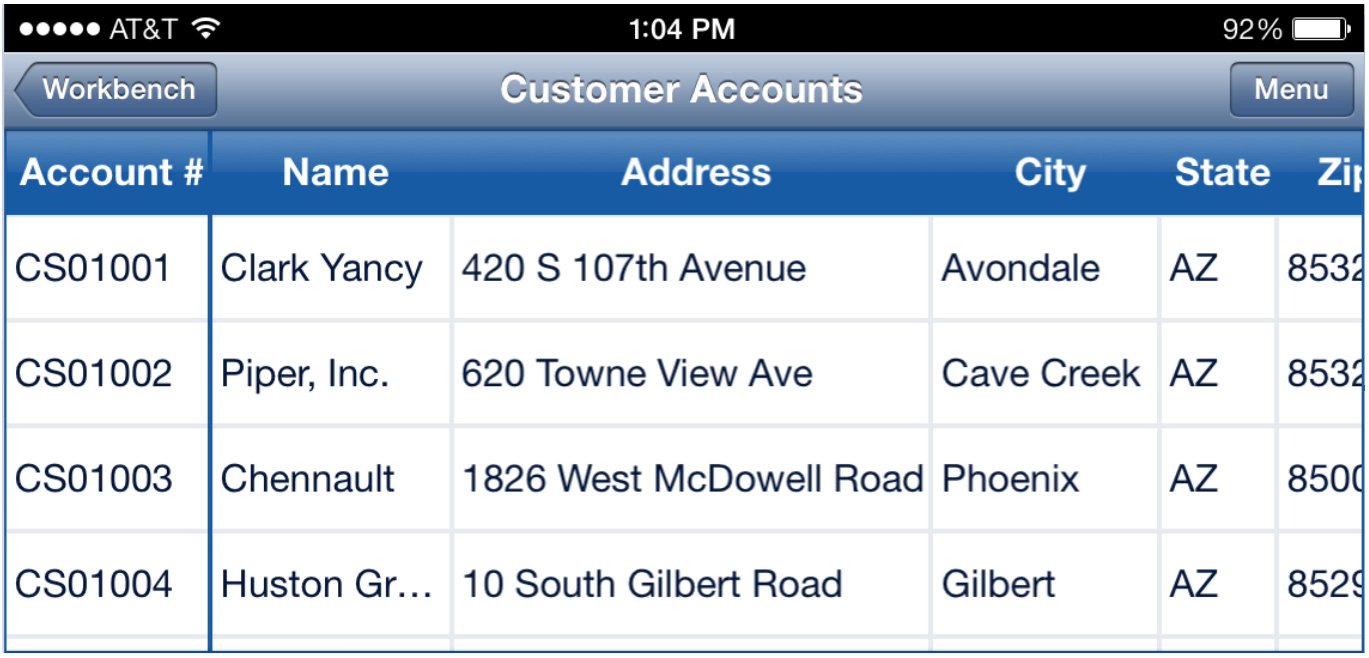
Account #: CS01002
Name: Piper, Inc.
Address: 620 Towne View Ave

Account #: CS01003
Name: Chennault
Address: 1826 West McDowell Road

Account #: CS01004
Name: Huston Group
Address: 10 South Gilbert Road

Account #: CS01005

Figure 12: An example of a mobile Query in Form style

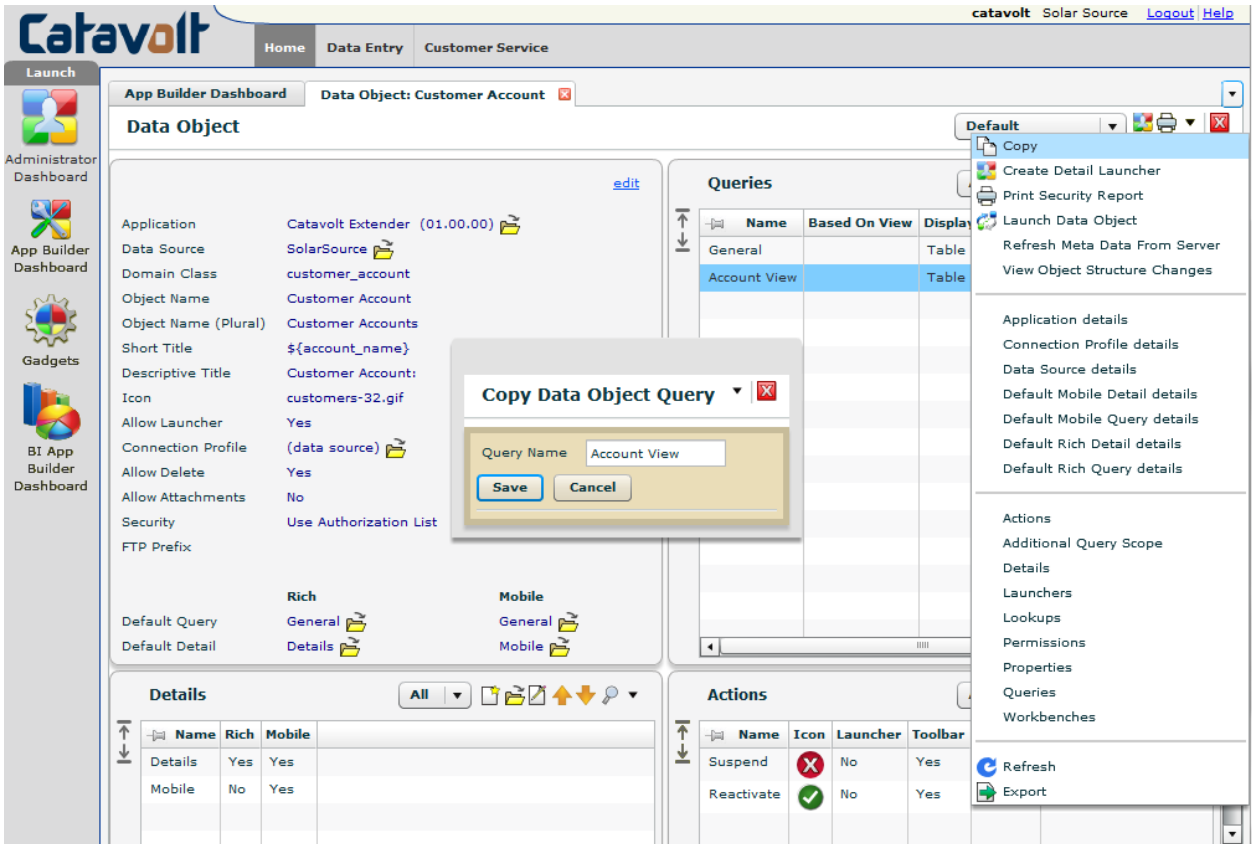


Account #	Name	Address	City	State	Zi
CS01001	Clark Yancy	420 S 107th Avenue	Avondale	AZ	8532
CS01002	Piper, Inc.	620 Towne View Ave	Cave Creek	AZ	8532
CS01003	Chennault	1826 West McDowell Road	Phoenix	AZ	850
CS01004	Huston Gr…	10 South Gilbert Road	Gilbert	AZ	852

Figure 13: An example of a mobile Query in Grid style

Copying Queries

You may have instances where you need to make a copy of an existing Query to show a different set of properties, or to render them in a different way. You can select the Copy menu option to accomplish this. When Copying a Query, you will be prompted to supply the new Query Name. An exact copy of this Query along with all of its components will be made.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Short Title	${account_name}
Descriptive Title	Customer Account:
Icon	customers-32.gif
Allow Launcher	Yes
Connection Profile	(data source)
Allow Delete	Yes
Allow Attachments	No
Security	Use Authorization List
FTP Prefix	

	Rich	Mobile
Default Query	General	General
Default Detail	Details	Mobile

Queries

Name	Based On View	Display
General		Table
Account View		Table

Copy Data Object Query

Query Name: Account View

Details

Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

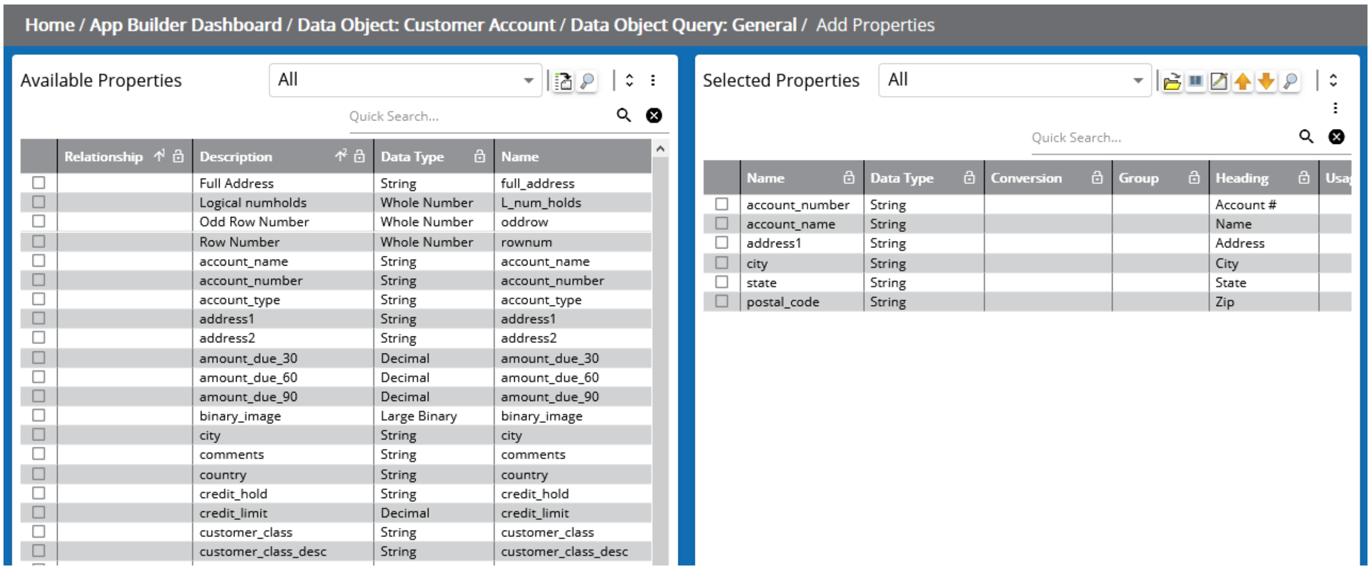
Actions

Name	Icon	Launcher	Toolbar
Suspend	✗	No	Yes
Reactivate	✓	No	Yes

Figure 14: The copy Data Object Query prompt

View Properties

View Properties specifies which properties should appear in the query and the order they should appear in. When adding View Properties, you will be presented with two lists. The Available Properties list shows all properties for the Data Object (including properties from related foreign keys and Additional Query Scopes). The Selected Properties list shows the properties that will be displayed on the Query



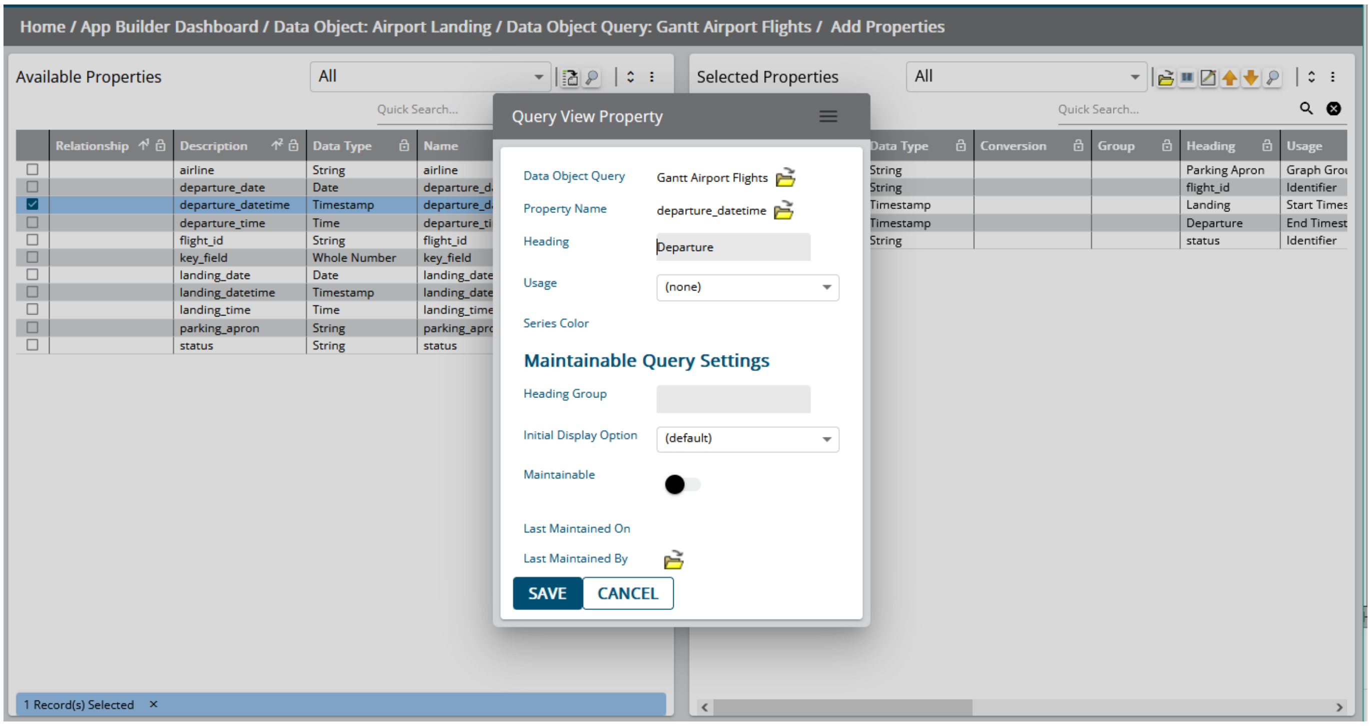
Home / App Builder Dashboard / Data Object: Customer Account / Data Object Query: General / Add Properties

	Relationship	Description	Data Type	Name
□		Full Address	String	full_address
□		Logical numholds	Whole Number	L_num_holds
□		Odd Row Number	Whole Number	oddrow
□		Row Number	Whole Number	rownum
□		account_name	String	account_name
□		account_number	String	account_number
□		account_type	String	account_type
□		address1	String	address1
□		address2	String	address2
□		amount_due_30	Decimal	amount_due_30
□		amount_due_60	Decimal	amount_due_60
□		amount_due_90	Decimal	amount_due_90
□		binary_image	Large Binary	binary_image
□		city	String	city
□		comments	String	comments
□		country	String	country
□		credit_hold	String	credit_hold
□		credit_limit	Decimal	credit_limit
□		customer_class	String	customer_class
□		customer_class_desc	String	customer_class_desc

	Name	Data Type	Conversion	Group	Heading	Usa
□	account_number	String			Account #	
□	account_name	String			Name	
□	address1	String			Address	
□	city	String			City	
□	state	String			State	
□	postal_code	String			Zip	

Figure 15: The Add Query View Properties view

You can select a single or multiple properties and press the Add Properties button to add them to the Query. If you select a single property, you will be presented with the following dialog:



Home / App Builder Dashboard / Data Object: Airport Landing / Data Object Query: Gantt Airport Flights / Add Properties

Available Properties — All

	Relationship	Description	Data Type	Name
□		airline	String	airline
□		departure_date	Date	departure_d
☑		departure_datetime	Timestamp	departure_d
□		departure_time	Time	departure_ti
□		flight_id	String	flight_id
□		key_field	Whole Number	key_field
□		landing_date	Date	landing_date
□		landing_datetime	Timestamp	landing_date
□		landing_time	Time	landing_time
□		parking_apron	String	parking_aprc
□		status	String	status

1 Record(s) Selected

Selected Properties — All

Data Type	Conversion	Group	Heading	Usage
String			Parking Apron	Graph Grou
String			flight_id	Identifier
Timestamp			Landing	Start Times
Timestamp			Departure	End Timest
String			status	Identifier

Query View Property

Data Object Query: Gantt Airport Flights

Property Name: departure_datetime

Heading: Departure

Usage: (none)

Series Color

Maintainable Query Settings

Heading Group

Initial Display Option: (default)

Maintainable

Last Maintained On

Last Maintained By

Figure 16: The add Query View Property prompt for a single record selection

Heading specifies the column heading that will be displayed for the property. Xalt defaults this value to the Property's Description as specified in the Defined Properties section of the Data Object.

Usage specifies any special role this property is to play in the Query. The list of available usages is dependent on the type of property and the type of Query being displayed:

- Table – Image/Video, Sum, Min, Max
- Image Viewer – Identifier, Image/Video
- Calendar – Identifier, Start Date, Start Time, End Date, End Time, Occur Date, Occur Time
- Graph – Identifier, Graph Filter, Graph Group, Graph Value, Graph Value (Sum), Graph Value (Min), Graph Value (Max), Graph Value (X-Axis), Start Date, Start Time, End Date, End Time, Start Timestamp, End Timestamp, and Bubble Size
- Map – Identifier, Street Address, City, State, Zip Code

Usages will be more fully described with their corresponding Query types.

Series Color specifies the color you wish to use to display the value with.

Maintainable specifies whether to make values in this property Maintainable when on a Gantt Chart. This value is only used for Start and End Date/Time/Timestamps. You must set both the Start and End values as maintainable to be able to drag/drop the Gantt bar to change the times.

The Maintainable Query Settings section contains values that are used when this Query is used as part of a Maintainable Query Dashboard. See Chapter 14: Dashboards for more information about creating a Maintainable Query Dashboard:

Heading Group specifies a second Heading value to be placed above the Heading. If multiple contiguous columns have the same Heading Group value, a single Heading Group spanning all the columns will be displayed.

Initial Display Option specifies how to present the column when the Query is first displayed. The options are:

- (default) – No special options (this is the default)
- Pinned – Initially pin this column. You may specify any combination of columns to pin (i.e. the columns are not required to be contiguous or the first columns in the query). At runtime, the client will initially move all pinned columns to the left of the query, then move them back to their original defined position when unpinned.
- First Visible – Horizontally scroll the list to make this column show as the first visible column. There can be only one column with this choice, so selecting this value will automatically unselect it for other columns in the same query. At runtime, the client will automatically horizontally "scroll" the list so that this is the first column showing after the pinned columns. If the user wishes to see the previous columns, they can manually scroll left. Note that once the user scrolls in either direction (left/right), the column scrolling becomes fully unlocked (i.e. if the user scrolls right, then scrolls all the way left, the first column in the query will be shown instead of the initial first "non-pinned" column).
- Hidden – Initially hide this column. The user can specify any combination of columns to hide (i.e. the columns are not required to be contiguous or the first columns in the query, however there must be at least 1 non-Hidden column in the Query). At runtime, the client will initially hide all specified columns. The user can choose to show any hidden columns.

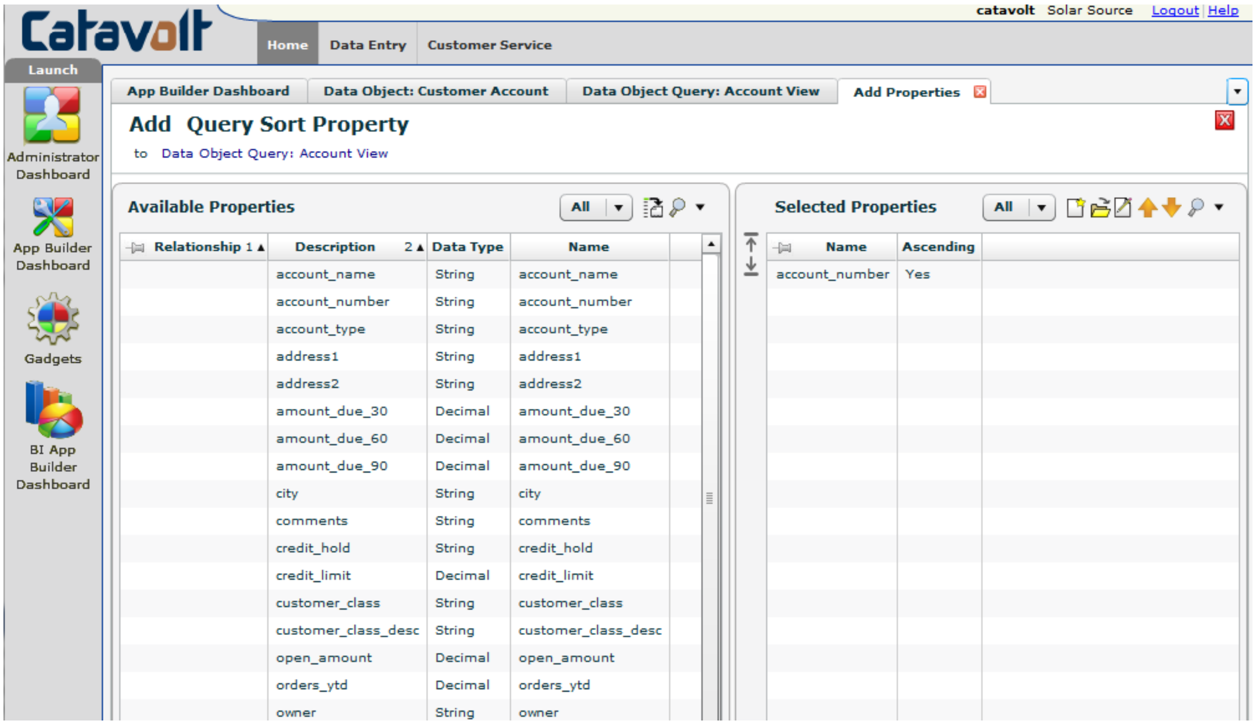
Maintainable specifies whether to make values in this column Maintainable when on a Maintainable query (default is No). This value is only available for simple String, Whole Number, Decimal, Date, Time, Timestamp, and Boolean properties. If the underlying property is using a drop-down list (Object Lookups, Property Values, etc), this value will enabled for simple drop-downs. If the underlying property is for a drop-down that is dependent on another property or drop-down to load its value, it will be unchecked and disabled. In addition, if the underlying property for this column is itself not maintainable, this value will be unchecked and disabled).

You can get to the same dialog after adding the property to a Property Section by selecting it in the Selected Properties list and pressing the Edit button.

There is an **Add Maintainable Query Spacer Column** menu option that will create a (blank) spacer column when this Query is used as part of a Maintainable Query Dashboard. Spacer Columns will be ignored when used as a regular Query. See Chapter 14: Dashboards for more information about creating a Maintainable Query Dashboard.

Sort Properties

Sort Properties specifies which properties should be used to sort the record of the Query. You can specify multiple sort fields. If two records contain the same value for the first Sort Property, they will be compared against the second Sort Property, and so on. When adding Sort Properties, you will be presented with two lists. The Available Properties list shows all properties for the Data Object (including properties from related foreign keys and Additional Query Scopes). The Selected Properties list shows the properties that will be used to sort the Query



Add Query Sort Property

to Data Object Query: Account View

Available Properties

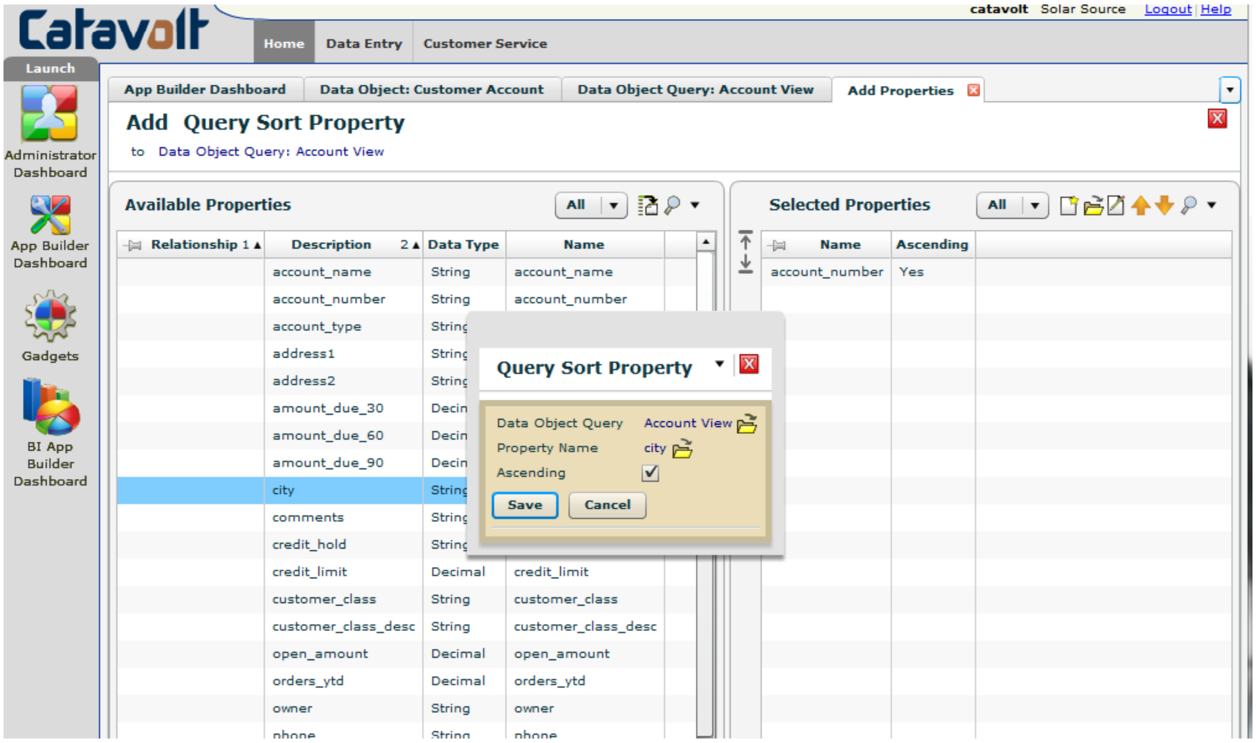
Relationship 1 ▲	Description 2 ▲	Data Type	Name
	account_name	String	account_name
	account_number	String	account_number
	account_type	String	account_type
	address1	String	address1
	address2	String	address2
	amount_due_30	Decimal	amount_due_30
	amount_due_60	Decimal	amount_due_60
	amount_due_90	Decimal	amount_due_90
	city	String	city
	comments	String	comments
	credit_hold	String	credit_hold
	credit_limit	Decimal	credit_limit
	customer_class	String	customer_class
	customer_class_desc	String	customer_class_desc
	open_amount	Decimal	open_amount
	orders_ytd	Decimal	orders_ytd
	owner	String	owner

Selected Properties

Name	Ascending
account_number	Yes

Figure 17: The Add Query Sort Properties view

You can select a single or multiple properties and press the Add Properties button to add them to the Sort. If you select a single property, you will be presented with the following dialog:



Add Query Sort Property

to Data Object Query: Account View

Available Properties

Relationship 1 ▲	Description 2 ▲	Data Type	Name
	account_name	String	account_name
	account_number	String	account_number
	account_type	Strin	
	address1	Strin	
	address2	Strin	
	amount_due_30	Decin	
	amount_due_60	Decin	
	amount_due_90	Decin	
	city	Strin	
	comments	Strin	
	credit_hold	Strin	
	credit_limit	Decimal	credit_limit
	customer_class	String	customer_class
	customer_class_desc	String	customer_class_desc
	open_amount	Decimal	open_amount
	orders_ytd	Decimal	orders_ytd
	owner	String	owner
	phone	String	phone

Selected Properties

Name	Ascending
account_number	Yes

Query Sort Property

Data Object Query: Account View
Property Name: city
Ascending: ☑

Save | Cancel

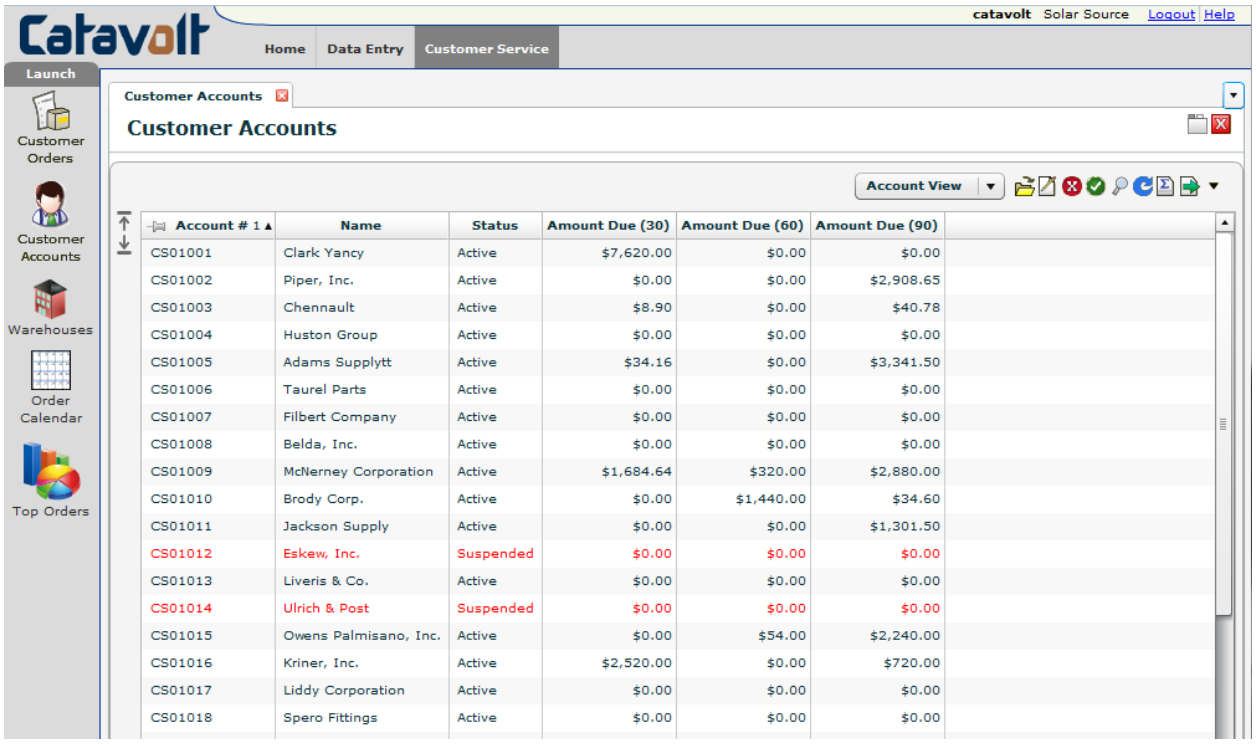
Figure 18: The add Query Sort Property view

When creating a Query Sort Property, you must specify a Property Name. Clicking on the find button will bring up a list of all properties for the Data Object (including properties from related foreign keys and Additional Query Scopes).

Ascending specifies whether the records should be sorted in ascending (A-Z) or descending (Z-A) order. Selecting the checkbox specifies ascending (A-Z) order.

Data Annotations

Data Annotations allow you to highlight Query data using colors, fonts, images and alternate text. Data Annotations allow you to add background colors, foreground colors, bold lettering, italic lettering and override text (to either entire rows or cells within a row) based on the data being displayed. The end user sees the annotations in their lists, maps and calendars.

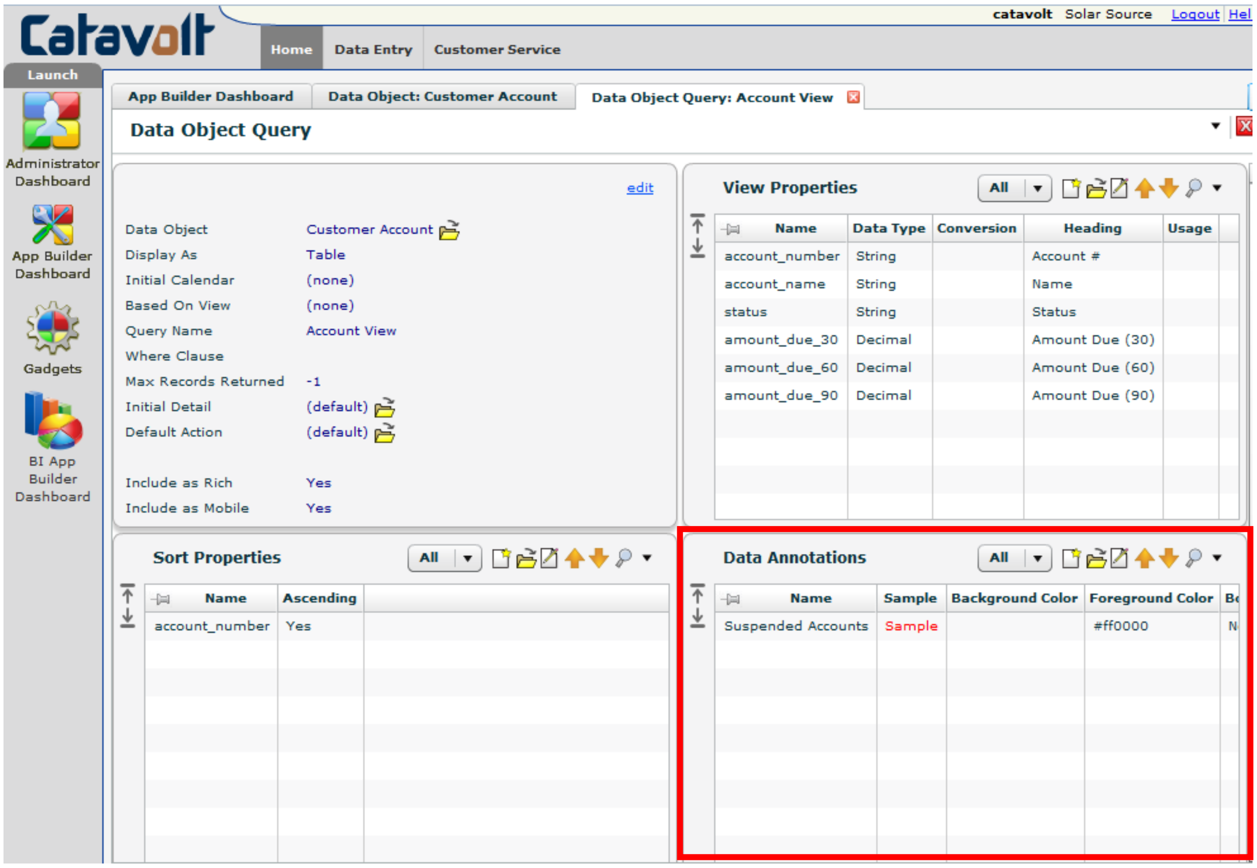


Customer Accounts

Account # 1 ▲	Name	Status	Amount Due (30)	Amount Due (60)	Amount Due (90)
CS01001	Clark Yancy	Active	$7,620.00	$0.00	$0.00
CS01002	Piper, Inc.	Active	$0.00	$0.00	$2,908.65
CS01003	Chennault	Active	$8.90	$0.00	$40.78
CS01004	Huston Group	Active	$0.00	$0.00	$0.00
CS01005	Adams Supplytt	Active	$34.16	$0.00	$3,341.50
CS01006	Taurel Parts	Active	$0.00	$0.00	$0.00
CS01007	Filbert Company	Active	$0.00	$0.00	$0.00
CS01008	Belda, Inc.	Active	$0.00	$0.00	$0.00
CS01009	McNerney Corporation	Active	$1,684.64	$320.00	$2,880.00
CS01010	Brody Corp.	Active	$0.00	$1,440.00	$34.60
CS01011	Jackson Supply	Active	$0.00	$0.00	$1,301.50
CS01012	Eskew, Inc.	Suspended	$0.00	$0.00	$0.00
CS01013	Liveris & Co.	Active	$0.00	$0.00	$0.00
CS01014	Ulrich & Post	Suspended	$0.00	$0.00	$0.00
CS01015	Owens Palmisano, Inc.	Active	$0.00	$54.00	$2,240.00
CS01016	Kriner, Inc.	Active	$2,520.00	$0.00	$720.00
CS01017	Liddy Corporation	Active	$0.00	$0.00	$0.00
CS01018	Spero Fittings	Active	$0.00	$0.00	$0.00

Figure 19: An example query with a data annotation highlighting suspended records in red

Multiple Data Annotations are allowed for a Query. Each Data Annotation can affect either one or more individual View Properties or the entire row. When multiple Data Annotations affect the same View Property or row, the last Annotation in the list will be applied. To change the order, select a Data Annotation and press the Move Up and Move Down toolbar buttons. Data Annotations are available for Tables, Maps, and Calendars.



Data Object Query

Data Object	Customer Account
Display As	Table
Initial Calendar	(none)
Based On View	(none)
Query Name	Account View
Where Clause	
Max Records Returned	-1
Initial Detail	(default)
Default Action	(default)
Include as Rich	Yes
Include as Mobile	Yes

View Properties

Name	Data Type	Conversion	Heading	Usage
account_number	String		Account #	
account_name	String		Name	
status	String		Status	
amount_due_30	Decimal		Amount Due (30)	
amount_due_60	Decimal		Amount Due (60)	
amount_due_90	Decimal		Amount Due (90)	

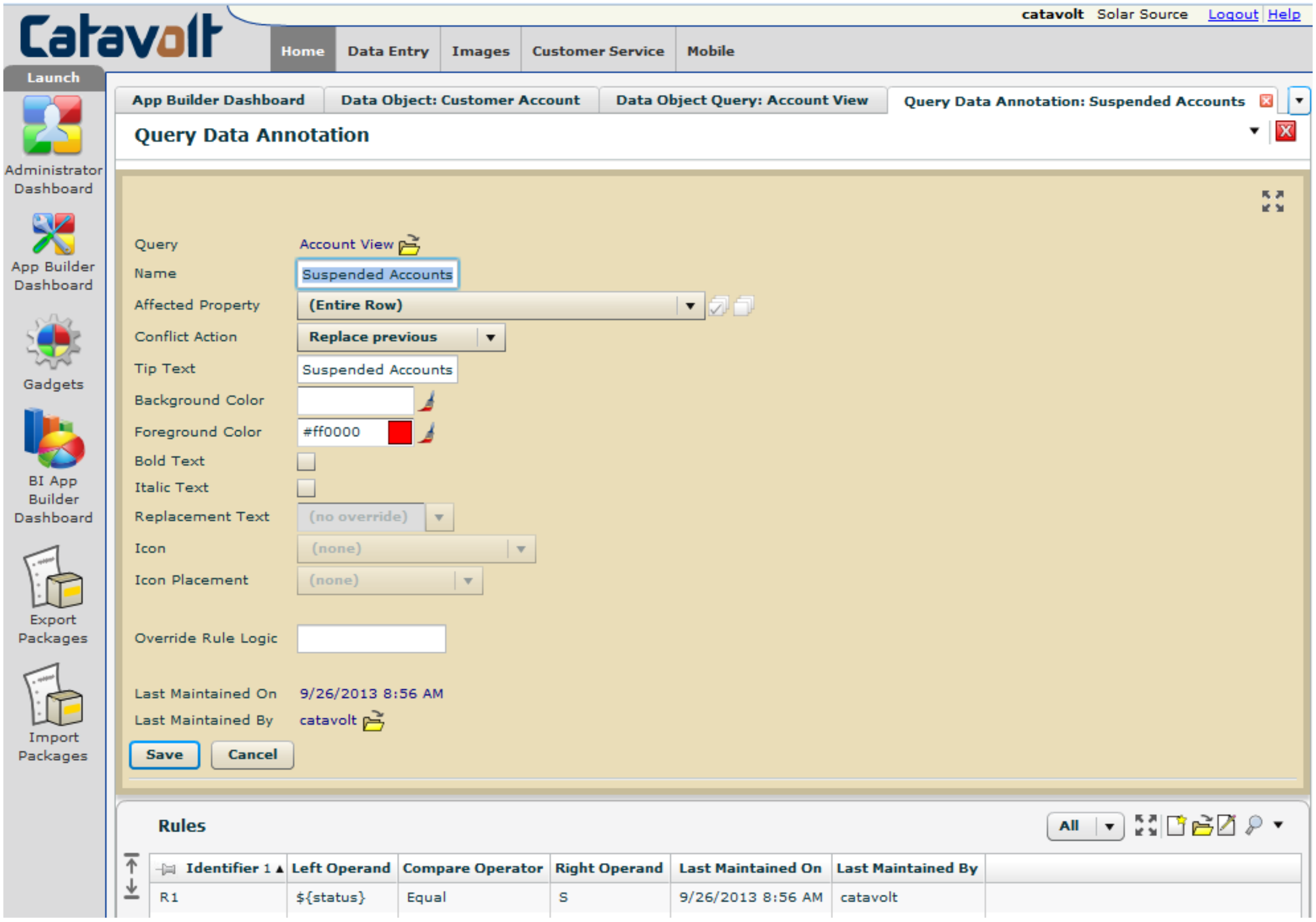
Sort Properties

Name	Ascending
account_number	Yes

Data Annotations

Name	Sample	Background Color	Foreground Color	Bo
Suspended Accounts	Sample		#ff0000	N

Figure 20: The Data Object Query details view highlighting the Data Annotation query section



Query Data Annotation

Query	Account View
Name	Suspended Accounts
Affected Property	(Entire Row)
Conflict Action	Replace previous
Tip Text	Suspended Accounts
Background Color	
Foreground Color	#ff0000 🟥
Bold Text	□
Italic Text	□
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)
Override Rule Logic	
Last Maintained On	9/26/2013 8:56 AM
Last Maintained By	catavolt

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand	Last Maintained On	Last Maintained By
R1	${status}	Equal	S	9/26/2013 8:56 AM	catavolt

Figure 21: The create Query Data Annotations view

When creating a Data Annotation, you must specify a Name. This value is used to uniquely identify the Data Annotation and is not presented to the end user.

Affected Property specifies which View Property(s) will be altered by the Data Annotation or if it applies to the whole record. Xalt defaults this value to (Entire Row), which will alter every View Property for the specified row. You also have the option of choosing one or more View Properties from the Query. Note that when creating a Data Annotation for a Calendar, (Entire Row) is the only available option.

Conflict Action specifies what should be done if multiple Data Annotations apply to a single row or a single property. As stated above, Annotations are processed in a specific sequence. Two annotations are in conflict with each other when they both pass all their rules for a row and are annotating the same property or the entire row. When annotations are in conflict, the last one in the list wins. This option allows the annotation to either be merged with previous annotations or replace previous annotations. When an annotation is merged with a previous annotation, it will assume all previous annotations' properties that it does not specify itself. For example, if one annotation sets the background color and a latter one specifies a foreground color and they are merged, the resulting annotation will have both a background and a foreground color set.

Tip Text specifies text that will be displayed when the end user hovers over a row or property currently affected by a Data Annotation. In this way, the end user can get a text description of the Data Annotation that is currently being applied to the row or property without having to remember what the different colors, fonts, etc. mean when looking at a Query.

Background Color specifies the color you wish to change the background of the affected cell or the entire row if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Foreground Color specifies the color you wish to change the foreground text of the affected cell or the entire row if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Bold Text specifies whether you want to change the font to be bold. Select the checkbox to apply a bold font to the affected cell or the entire row if the Data Annotation is applied.

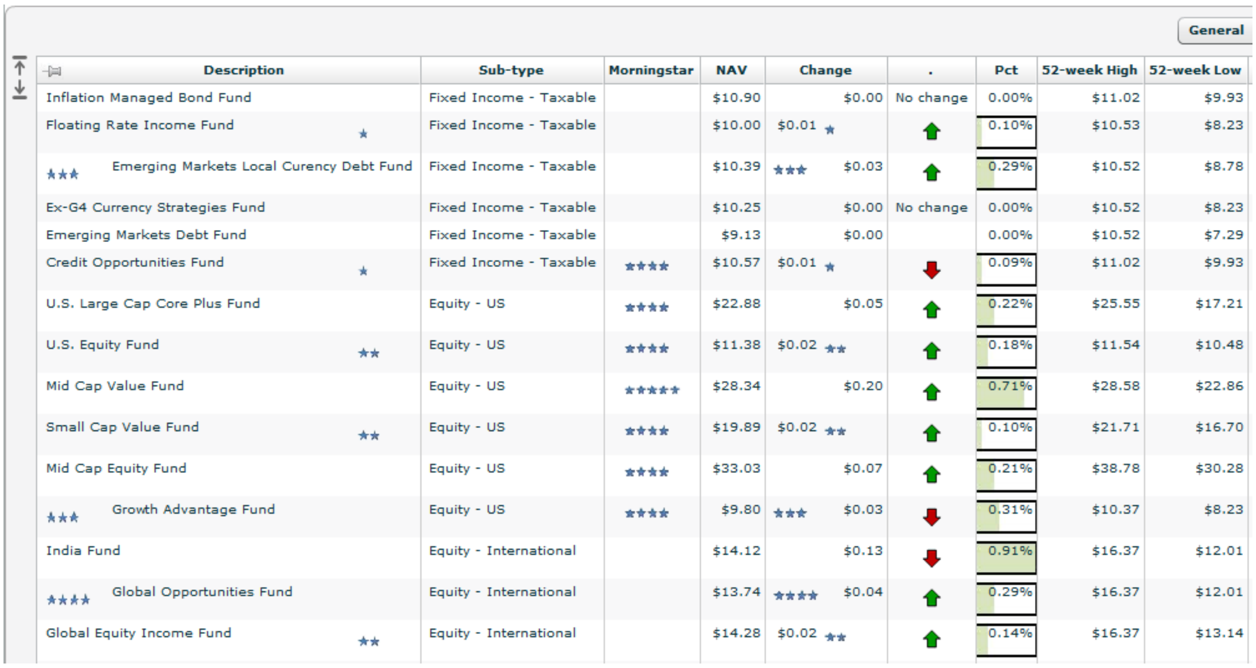
Italic Text specifies whether you want to change the font to be italicized. Select the checkbox to apply an italicized font to the affected cell or the entire row if the Data Annotation is applied.

Replacement Text specifies whether you want to replace the value of the affected cell with an alternate text string. This is commonly used to improve readability of the Query (for example, to replace 0 values with blanks). Note that Replacement Text is not available when the Affected Property is (Entire Row).

Icon specifies whether you want to append or replace the value of the affected cell with an image. The image you choose can be shown instead of or in addition to the value of the affected cell. Note that Icon is not available when the Affected Property is (Entire Row).

Icon Placement specifies where to place the image in relation to the value of the affected cell. The available values are:

- Replace Text – Replace the value of the affected cell with the image.
- Left of Text – Put the image to the left of the value of the affected cell.
- Right of Text – Put the image to the right of the value of the affected cell.
- Background – Put the image in the background under the value of the affected cell. This value is not used on mobile clients.
- Background (Fill) – Put the image in the background under the value of the affected cell and stretch the image to fill the column horizontally. This is typically used for annotations that show progress bars. This value is not used on mobile clients



Description	Sub-type	Morningstar	NAV	Change	.	Pct	52-week High	52-week Low
Inflation Managed Bond Fund	Fixed Income - Taxable		$10.90	$0.00	No change	0.00%	$11.02	$9.93
Floating Rate Income Fund ★	Fixed Income - Taxable		$10.00	$0.01 ★	⬆	0.10%	$10.53	$8.23
★★★ Emerging Markets Local Curency Debt Fund	Fixed Income - Taxable		$10.39	★★★ $0.03	⬆	0.29%	$10.52	$8.78
Ex-G4 Currency Strategies Fund	Fixed Income - Taxable		$10.25	$0.00	No change	0.00%	$10.52	$8.23
Emerging Markets Debt Fund	Fixed Income - Taxable		$9.13	$0.00		0.00%	$10.52	$7.29
Credit Opportunities Fund ★	Fixed Income - Taxable	★★★★	$10.57	$0.01 ★	⬇	0.09%	$11.02	$9.93
U.S. Large Cap Core Plus Fund	Equity - US	★★★★	$22.88	$0.05	⬆	0.22%	$25.55	$17.21
U.S. Equity Fund ★★	Equity - US	★★★★	$11.38	$0.02 ★★	⬆	0.18%	$11.54	$10.48
Mid Cap Value Fund	Equity - US	★★★★★	$28.34	$0.20	⬆	0.71%	$28.58	$22.86
Small Cap Value Fund ★★	Equity - US	★★★★	$19.89	$0.02 ★★	⬆	0.10%	$21.71	$16.70
Mid Cap Equity Fund	Equity - US	★★★★	$33.03	$0.07	⬆	0.21%	$38.78	$30.28
★★★ Growth Advantage Fund	Equity - US	★★★★	$9.80	★★★ $0.03	⬇	0.31%	$10.37	$8.23
India Fund	Equity - International		$14.12	$0.13	⬇	0.91%	$16.37	$12.01
★★★★ Global Opportunities Fund	Equity - International		$13.74	★★★★ $0.04	⬆	0.29%	$16.37	$12.01
Global Equity Income Fund ★★	Equity - International		$14.28	$0.02 ★★	⬆	0.14%	$16.37	$13.14

Figure 22: A Query showing different examples of Icon placements

Override Rule Logic specifies more advanced logic to evaluate the Data Annotation Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Data Annotation to be applied. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

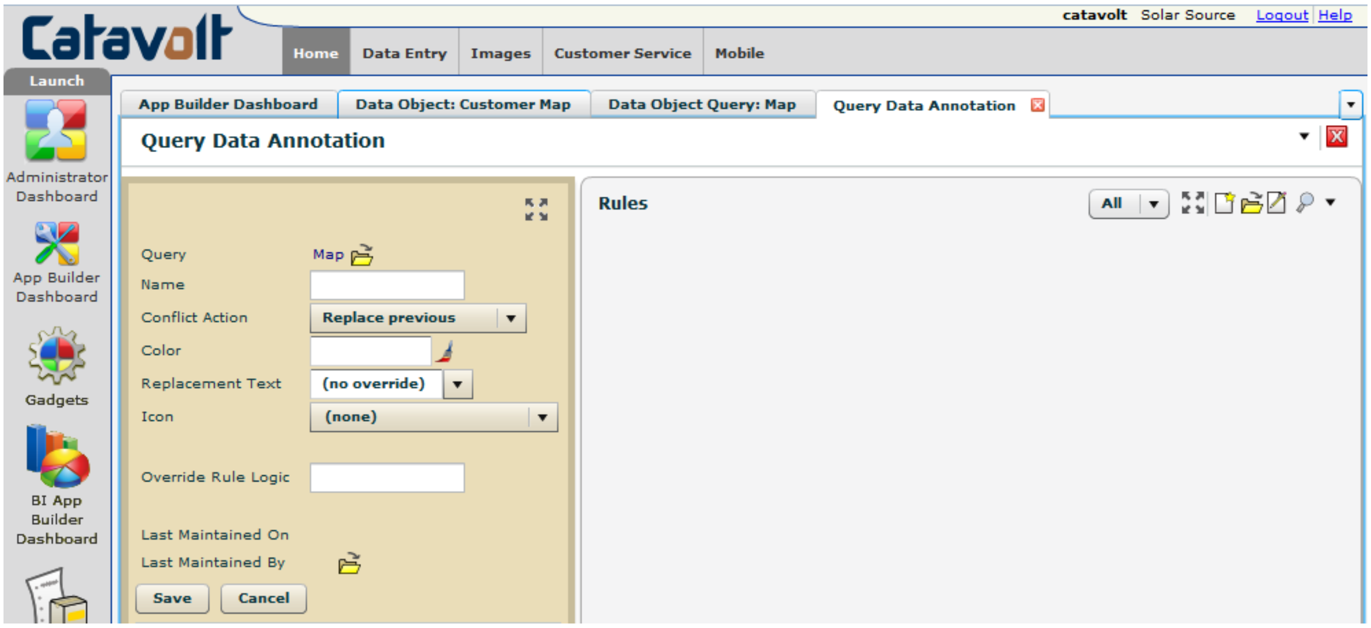
(R1 AND R2) OR (R3 AND R4)

NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

When creating a Data Annotation for a Map, the options are slightly different:



Query Data Annotation

Query	Map
Name	
Conflict Action	Replace previous
Color	
Replacement Text	(no override)
Icon	(none)
Override Rule Logic	
Last Maintained On	
Last Maintained By	

Figure 23: The create Query Data Annotations view for a Map Query

As before, you must specify a Name. This value is used to uniquely identify the Data Annotation and is not presented to the end user.

Conflict Action specifies what should be done if multiple Data Annotations apply to a single row or a single property. As stated above, Annotations are processed in a specific sequence. Two annotations are in conflict of each other when they both pass all their rules for a row and are annotating the same property or the entire row. When annotations are in conflict, the last one in the list wins. This option allows the annotation to either be merged with previous annotations or replace previous annotations. When an annotation is merged with a previous annotation, it will assume all previous annotations' properties that it does not specify itself. For example, if one annotation sets the background color and a latter one specifies a foreground color and they are merged, the resulting annotation will have both a background and a foreground color set.

Color specifies the color you wish to change the standard Map pin to. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Replacement Text specifies whether you want to replace the identifier of the standard Map pin with an alternate text string.

Icon specifies whether you want to replace the standard map pin image with a different image altogether.

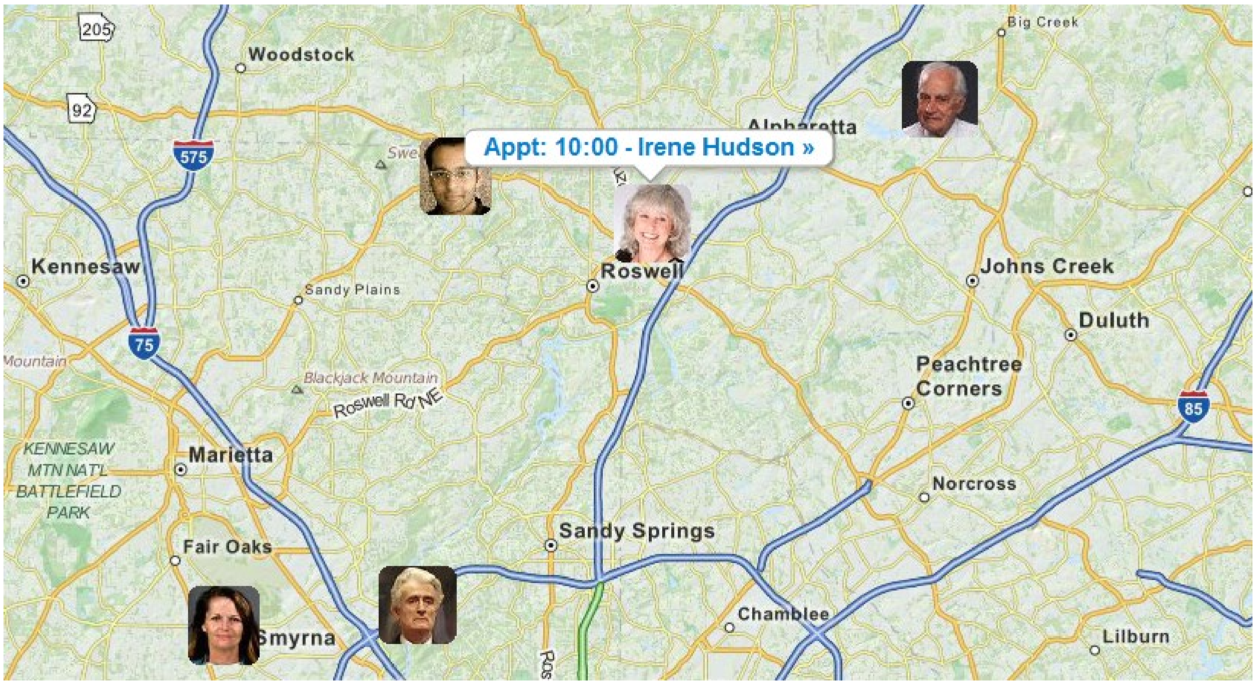


Figure 24: A Map showing Icons that have replaced the standard map pins

Override Rule Logic specifies more advanced logic to evaluate the Data Annotation Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Data Annotation to be applied. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

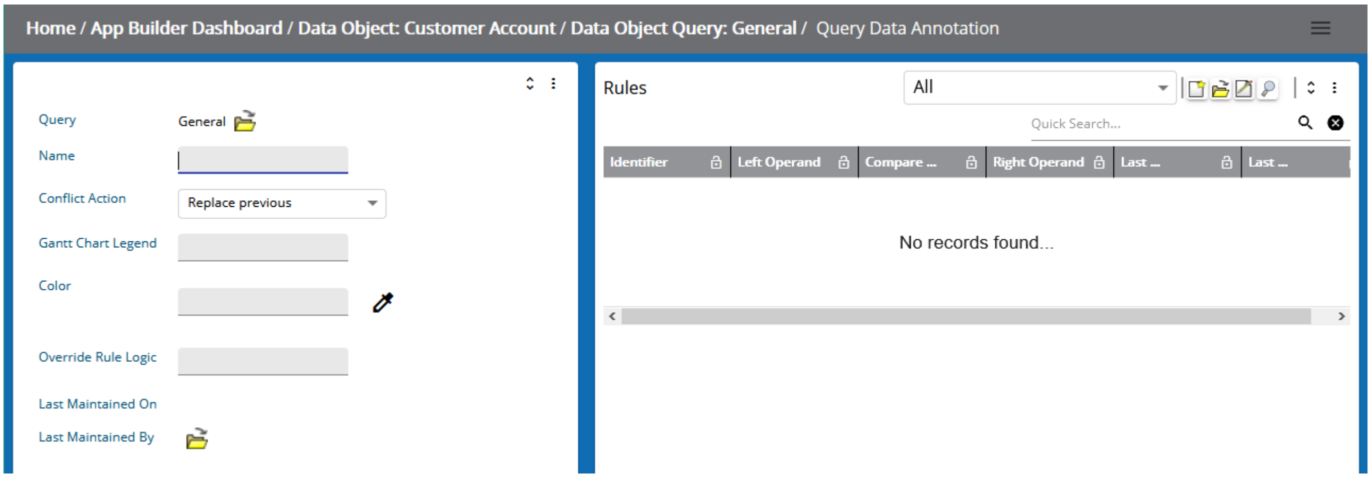
(R1 AND R2) OR (R3 AND R4)

NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

When creating a Data Annotation for a Graph, the options are also slightly different:



Home / App Builder Dashboard / Data Object: Customer Account / Data Object Query: General / Query Data Annotation

Query	General
Name	
Conflict Action	Replace previous
Gantt Chart Legend	
Color	
Override Rule Logic	
Last Maintained On	
Last Maintained By	

Rules — All

Quick Search...

Identifier	Left Operand	Compare ...	Right Operand	Last ...	Last ...

No records found...

Figure 25: The create Query Data Annotations view for a Graph Query

As before, you must specify a Name. This value is used to uniquely identify the Data Annotation and is not presented to the end user.

Conflict Action specifies what should be done if multiple Data Annotations apply to a single row or a single property. As stated above, Annotations are processed in a specific sequence. Two annotations are in conflict of each other when they both pass all their rules for a row and are annotating the same property or the entire row. When annotations are in conflict, the last one in the list wins. This option allows the annotation to either be merged with previous annotations or replace previous annotations. When an annotation is merged with a previous annotation, it will assume all previous annotations' properties that it does not specify itself. For example, if one annotation sets the background color and a latter one specifies a foreground color and they are merged, the resulting annotation will have both a background and a foreground color set.

Gantt Chart Legend specifies the text to display in the Legend for the specified Color.

Color specifies the color you wish to use to display the Graph Value. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged. This option is most useful on graphs that render objects instead of values, such as Pie Graphs, Donut Charts, Gantt Charts, Scatter Plots, and Bubble Charts.

Override Rule Logic specifies more advanced logic to evaluate the Data Annotation Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Data Annotation to be applied. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

(R1 AND R2) OR (R3 AND R4)

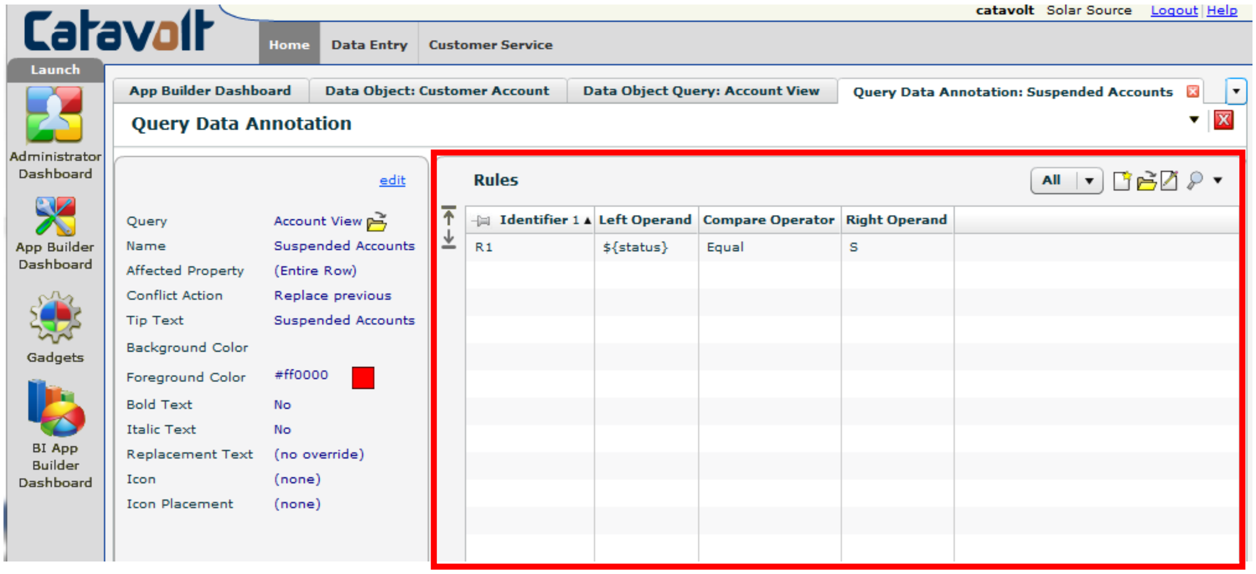
NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

Data Annotation Rules

The Rules list on a Data Annotation can be used to control when the Data Annotation should be applied to a row of data. A Data Annotation can have zero or more rules. All rules must pass for the Data Annotation to be applied to that particular row of data. If a Data Annotation has no rules, then it is considered to pass automatically.



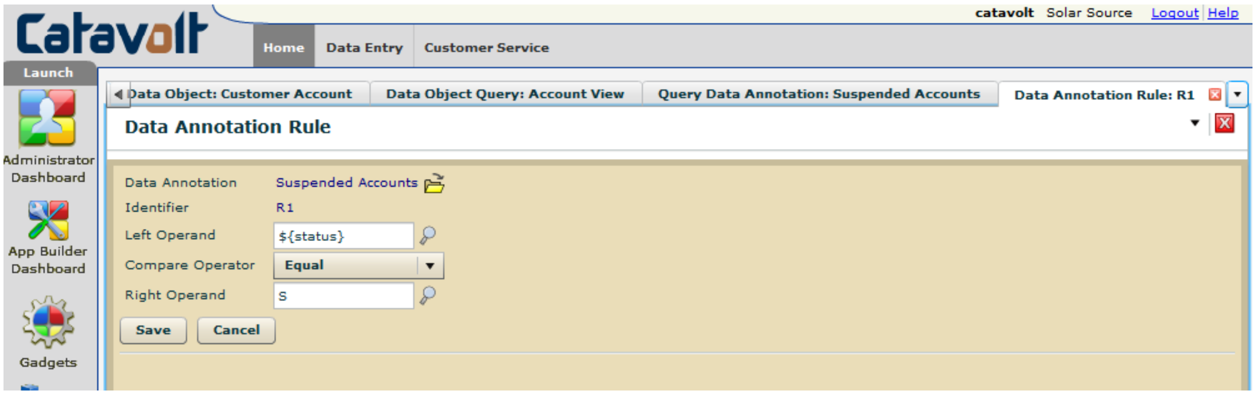
Query Data Annotation

Query	Account View
Name	Suspended Accounts
Affected Property	(Entire Row)
Conflict Action	Replace previous
Tip Text	Suspended Accounts
Background Color	
Foreground Color	#ff0000 🟥
Bold Text	No
Italic Text	No
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand
R1	${status}	Equal	S

Figure 26: Query Data Annotation details view with the Rules query section highlighted


Catavolt
catavolt Solar Source Logout | Help
Home
Data Entry
Customer Service
Launch
Administrator Dashboard
App Builder Dashboard
Gadgets
Data Object: Customer Account
Data Object Query: Account View
Query Data Annotation: Suspended Accounts
Data Annotation Rule: R1
Save
Cancel

Data Annotation Rule

Data Annotation	Suspended Accounts
Identifier	R1
Left Operand	${status}
Compare Operator	Equal
Right Operand	S

Figure 27: The create Data Annotation Rule view

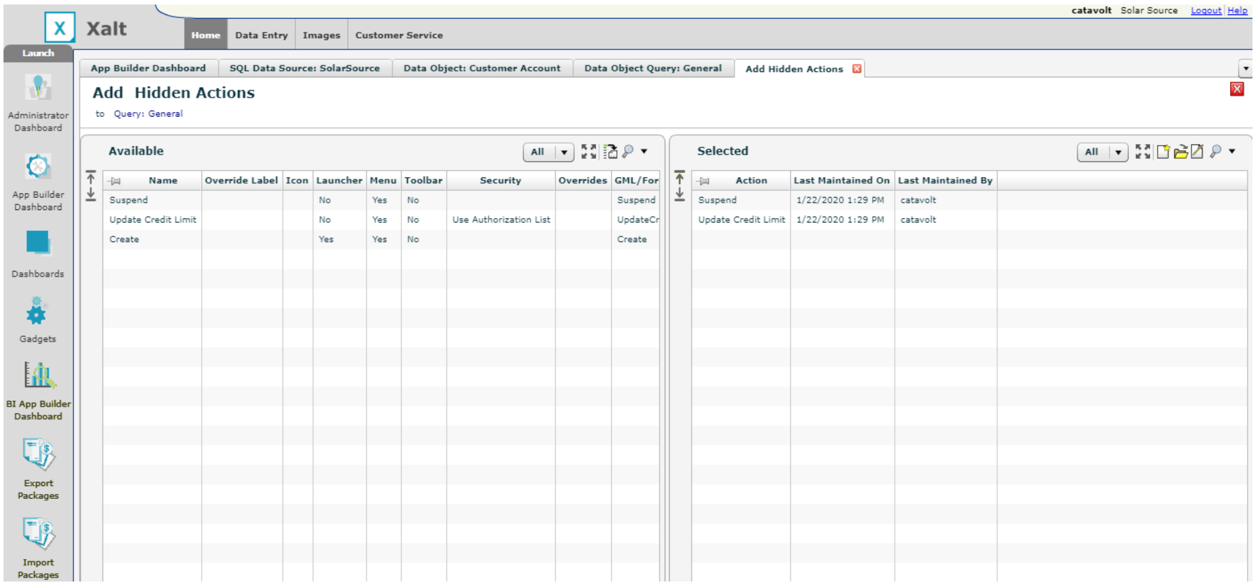
When creating a Data Annotation Rule, Xalt will automatically create a unique Identifier for the Rule.

Left Operand and Right Operand specify the two values that will be compared with each other. You have 2 choices for operands: Constant Values (such as A or 1.00) or Substitution Values. You can enter a Substitution Value manually or press the Find button to have Xalt build a Substitution Value based on the Available Properties for the Data Object. Appendix A: Specifying Messages and Substitution Values for more information about how to create Substitution Messages for Left Operand and Right Operand. A special Substitution Value of ${AFFECTED_PROPERTY} is available for annotations where you select multiple Affected Properties. When running the annotation, the rule will be checked independently for each affected property. This allows you to create a single Annotation and a single rule that can apply to multiple individual properties. For example, if you have 5 numeric columns in a Query and you want any negative values to show as red, you can create a single annotation to make the text foreground color red, choose all 5 numeric columns as Affected Properties, and create a single Annotation Rule of ${AFFECTED_PROPERTY} Less Than 0.

Compare Operator specifies how the two operands should be compared with each other. Options include Equal, Not Equal, Greater Than, Less Than, Greater or Equal, Less or Equal, Contains, Starts With, Ends With, Is Null, and Is Not Null.

Hidden Actions

Hidden Actions specifies a list of Actions that should not be shown on the menu when this Query is active (if you have set the Menu Actions property to Use Hidden Actions List). When adding Hidden Actions, you will be presented with two lists. The Available Actions list shows all actions for the Data Object (including both custom actions as well as override standard actions). The Selected Actions list shows the actions that will be hidden on the menu.



Add Hidden Actions

to Query: General

Available

Name	Override Label	Icon	Launcher	Menu	Toolbar	Security	Overrides	GML/For
Suspend			No	Yes	No			Suspend
Update Credit Limit			No	Yes	No	Use Authorization List		UpdateCr
Create			Yes	Yes	No			Create

Selected

Action	Last Maintained On	Last Maintained By
Suspend	1/22/2020 1:29 PM	catavolt
Update Credit Limit	1/22/2020 1:29 PM	catavolt

Figure 28: The Add Hidden Actions view

Excluding an Action from the menu will mark the action as Hidden. It can still be called directly from GML, same as with Hidden actions. It can also be used as a Default Action in a Query (action to be performed when a record is double-clicked).

The available list of Hidden Actions includes all custom actions in the Action list for the Data Object. If the you want to exclude a standard action (such as Export, Refresh, etc.), you should first run Override Standard Action and add it to the Actions list. At that point the Action will be available to add to the Hidden Actions list.

Since a menu is tied to a specific Query/Detail, if you have a Detail Query Section / Dashboard Query Section / Open List Completion Action and you want to have a special menu just for that section, you should create a dedicated (hidden) Query just to be used for that section that contains a custom menu.

Creating Table Queries

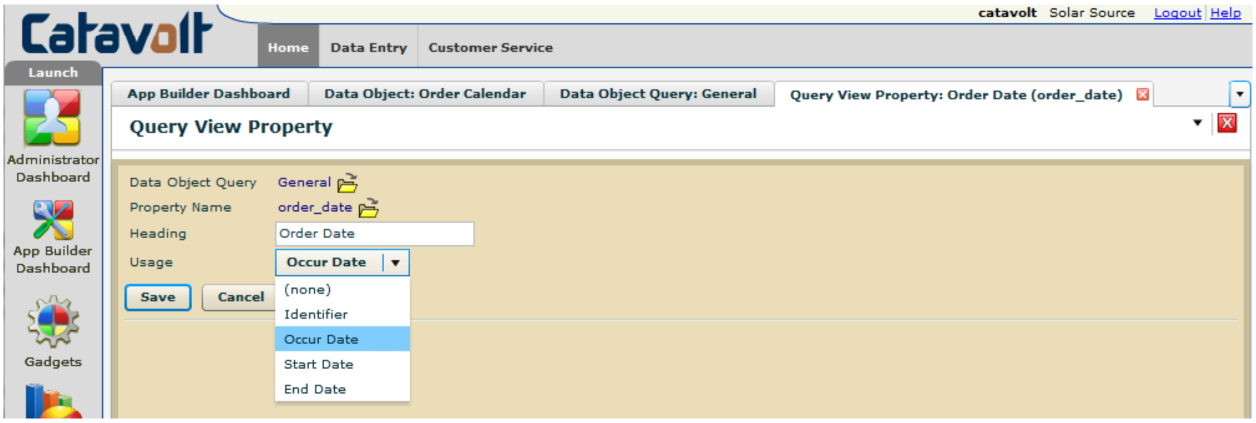
To create a Table Query, follow the instructions outlined above in Creating Queries. Set the Display As option to Table. If you have a URL property that references an image and you would like to display the image in the table, set the Usage to Image/Video for that property. For Numeric properties in Direct SQL and IBM i Data Sources, you may also choose a Usage of Sum, Min, or Max. This will turn your normal Query into a Summary (Group-By) Query. There are special considerations you need to know when running Summary Queries. Please see Appendix C: Summary (Group-By) Queries for more information.

Creating Calendar Queries

To create a Calendar Query, follow the instructions outlined above in Creating Queries. Set the Display As option to Calendar. The Initial Calendar property will default to Month View. You can change this to Day View or Week View to have the calendar initially display over these time periods.

When displaying a record on a calendar Xalt needs to know the date or dates that the record represents, the time that the record represents [if any] and an identifier for the text that will be shown in the calendar view.

For a Calendar Query you need to identify either an Occur Date or a Start Date and End Date. Add the Occur Date / Start Date / End Date properties to the Query as View Properties, setting the appropriate Usage.



Query View Property

Data Object Query General
Property Name order_date
Heading Order Date
Usage Occur Date

Figure 29: Query View Property details for a calendar Query in edit mode with the Usage dropdown shown.

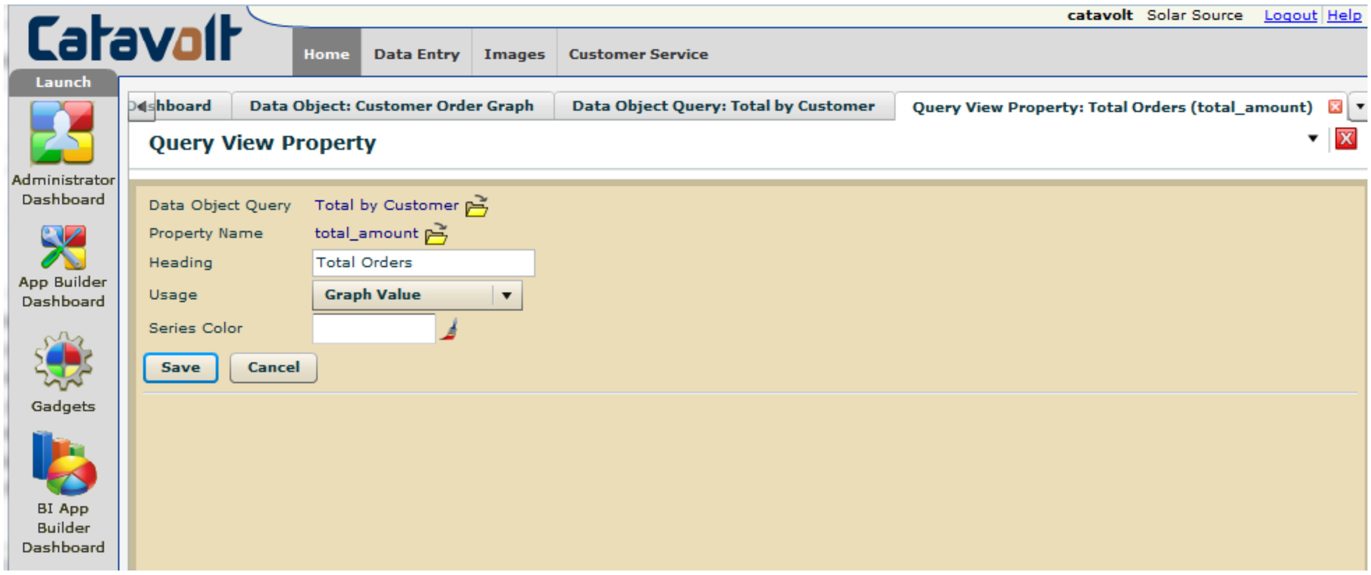
If your Data Object contains Time properties as well as Dates, you can also set the Usage for these properties to Occur Time, Start Time, and End Time. Identifying times is useful when displaying the Calendar in Day view.

You can also set the Usage for one or more View Properties to Identifier. The Calendar will take all the Identifier View Properties and use them when displaying the record on the Calendar. If there are multiple Identifier View Properties, they will be appended together with comma separators. If you specify no Identifier properties, Xalt will use the Data Object's Short Title when showing the record in the Calendar.

Creating Graph Queries

To create a Graph Query, follow the instructions outlined above in Creating Queries. Set the Display As option to either Bar Graph, Bubble Chart, Donut Chart, Gantt Chart, Line Graph, Pie Graph, Scatter Plot, or Stacked Bar Graph.

You need to identify a View Property as the Graph Value. The Graph Value is a Numeric property that contains the value to be graphed (Order amount, Days late, etc.). Add the property to the Query as a View Property, setting the Usage to Graph Value. You can specify multiple Graph Value properties to show more than one set of values in your graph. The Heading for the Query View Properties that are used as graph values will be used in the legend or tip text of the graph. For Numeric properties in Direct SQL and IBM i Data Sources, you may also choose a Usage of Graph Value (Sum), Graph Value (Min), or Graph Value (Max). This will turn your normal Query into a Summary (Group-By) Query. There are special considerations you need to know when running Summary Queries. Please see Appendix C: Summary (Group-By) Queries for more information. You can also override the color that will be used to render the graph value by setting a Series Color. Note that Graph Value and Series Color are not available for Gantt Charts.


Catavolt
catavolt Solar Source Logout | Help
Home
Data Entry
Images
Customer Service
Launch
Administrator Dashboard
App Builder Dashboard
Gadgets
BI App Builder Dashboard
Dashboard
Data Object: Customer Order Graph
Data Object Query: Total by Customer
Query View Property: Total Orders (total_amount)
Save
Cancel

Query View Property

Data Object Query: Total by Customer
Property Name: total_amount
Heading: Total Orders
Usage: Graph Value
Series Color:

Figure 30: Query View Property details for a graph Query in edit mode with the Usage drop down shown.

You can optionally identify a View Property as the Graph Filter. Xalt will build a filtering control that is based on all distinct values in the graph. For numeric properties the filtering control will be a range slider that ranges between the smallest value in the graph to the largest value. For date properties the filtering control will be from- and to- date fields that have the earliest and latest dates. For all other properties the filtering control will consist of check boxes for each distinct value in the graph data. As you select and unselect each checkbox, Xalt will automatically include/exclude records that contain that value from the graph. To enable graph filtering add the property to the

Query's View Properties, setting the Usage to Graph Filter. The View Property's heading will be used as the title of the filter control. You can add multiple Graph Filter properties. Note that Graph Filters are not available for Gantt Charts.

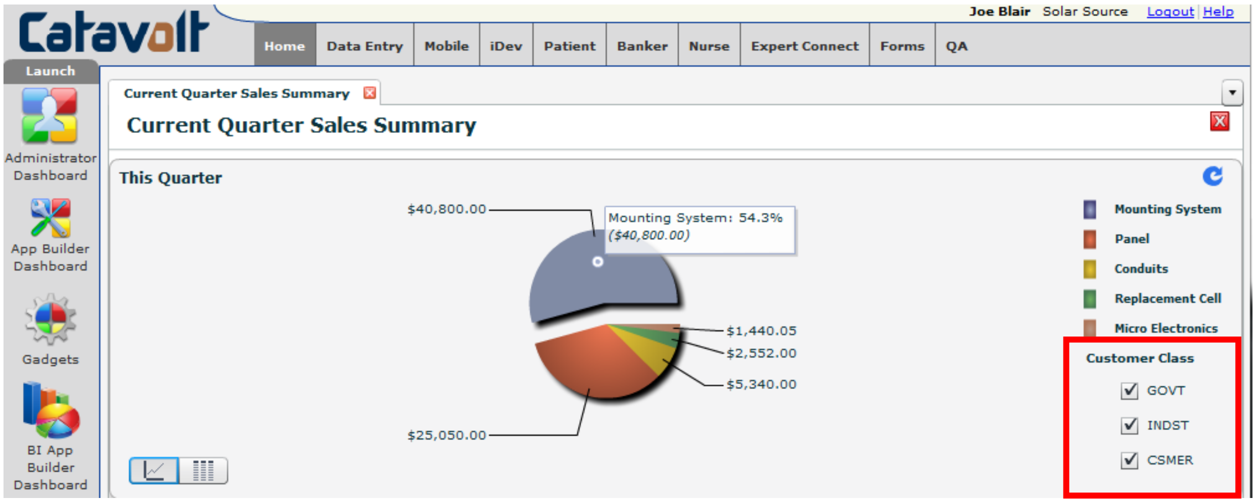


Figure 31: An example Graph Query with a filter control for a View Property named Customer Class

You can optionally identify a View Property as the Graph Group. Xalt will combine all records that have the same value for this property into a group that will be displayed and totaled together. For Gantt Charts, the Graph Group will be used to group individual items into lanes.

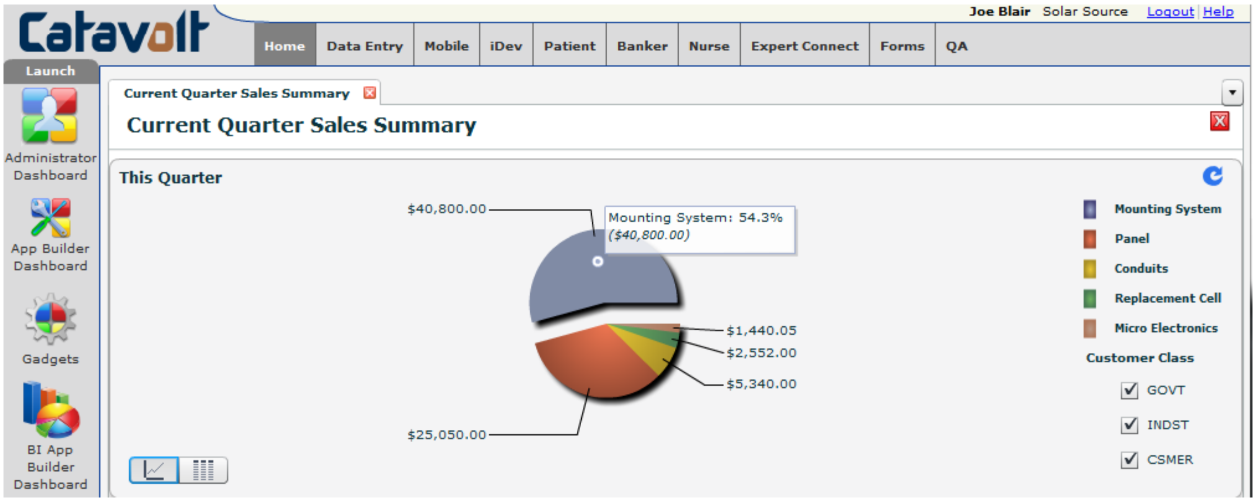


Figure 32: An example Graph Query shown with grouping where each pie slice represents a group of records

Double-clicking on the Group will take you to a second-level graph that shows the individual records making up the group. The second level graph will always show as a bar graph regardless of the top-level graph. The drill down graph will show the group name and a back button to get back to the grouped graph view. To use graph grouping add the property to the Query's View Properties, setting the Usage to Graph Group. Note that Graph Group is not available when using Bubble Charts and Scatter Plots, nor is it available for Summary (Group-By) queries, as the Summary query is already grouped. For Gantt Charts, double-clicking a bar will take you to the regular Details for that item.

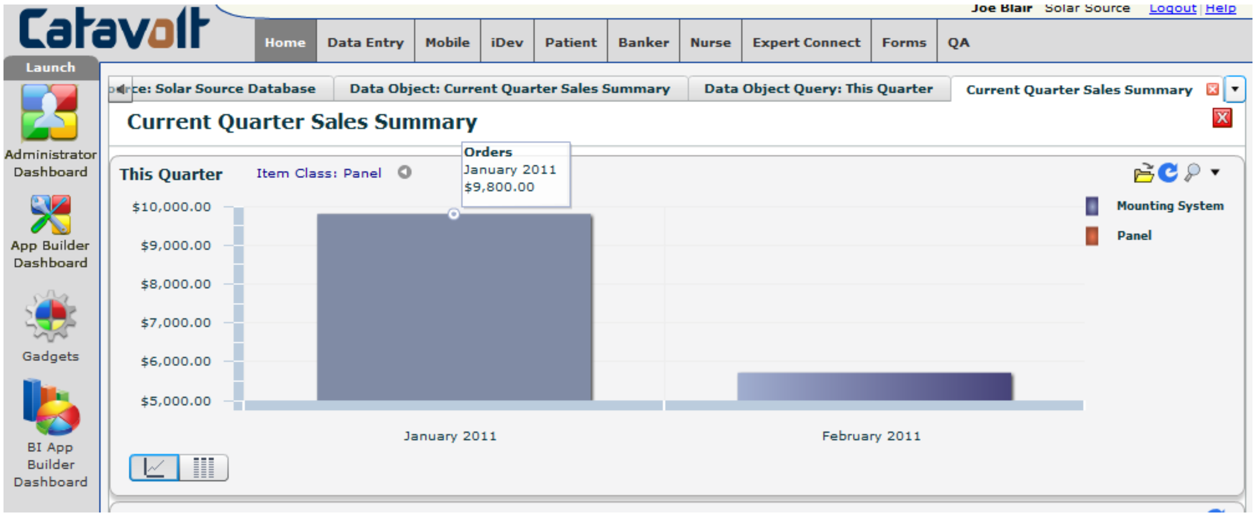


Figure 33: A drill down graph showing the records in a group

As in Calendars, you can also set the Usage for one or more View Properties to Identifier. The Graph will take all the Identifier View Properties and use them when displaying the record on the Graph. If there are multiple Identifier View Properties, they will be appended together with comma separators. If you specify no Identifier properties, Xalt will use the Data Object's Short Title when showing the record in the Graph.

Scatter Plots and Bubble Charts require both X and Y values in order to be graphed correctly. A Usage of Graph Value identifies the Y-Axis value (as it does on other graph types). In addition, you can specify a Usage of Graph Value (X-Axis) to identify the X-Axis value. Finally, Bubble Charts also utilize a property to determine how large to render each bubble, which can be identified by specifying a Usage of Bubble Size. If you define a Bubble Chart and do not specify a Bubble Size property, all bubbles will be given the same size.

Gantt Charts require a Start Date / Start Time, End Date / End Time or Start Timestamp / End Timestamp in order to place the bar on the graph. Add the Start Date / Start Time / End Date / End Time / Start Timestamp / End Timestamp properties to the Query as View Properties, setting the appropriate Usage. In addition, if you set the Maintainable property to Yes for both the Start and End properties, the user will be allowed to drag/drop an individual Gantt bar to change the date/time of the item.

Creating Map Queries

To create a Map Query, follow the instructions outlined above in Creating Queries. Set the Display As option to Map.

When displaying a record on a map Xalt needs to know the address the record represents and an identifier for the text that will be shown in the map view. An address can be specified in two ways. You can specify the combination of a longitude and latitude properties. Or, you can specify the combination of a Street, City, State and Postal Code properties. When specifying Street, City, State and Postal Code Xalt will use a geo-location service to retrieve the longitude and latitude values for that street address.

Add the address properties to the Query's View Properties and set the appropriate Usage for each property. You can also add one or more properties with a Usage of Identifier. Identifier properties will be concatenated together using a comma as a separator to identify a record on the map. If no Identifier property exits then the Data Object's short title will be used.

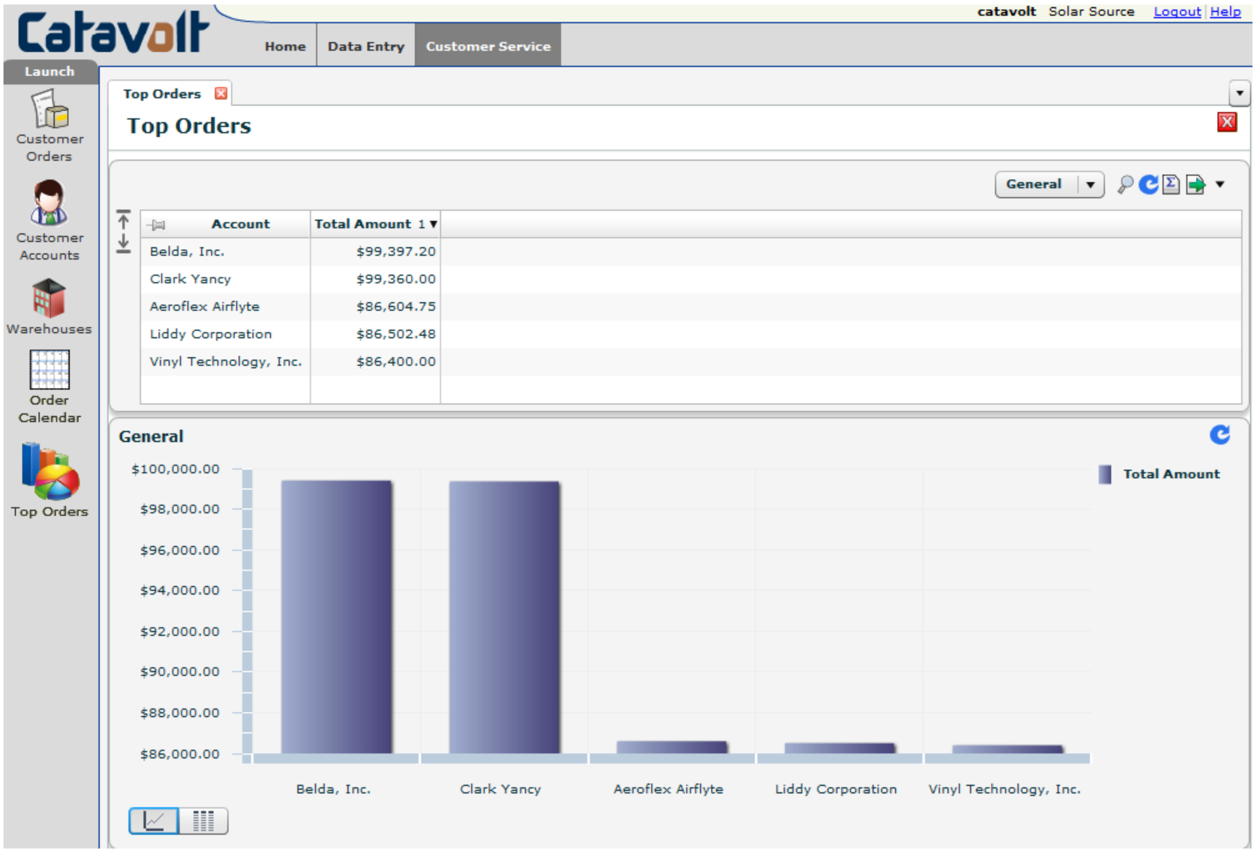
Creating Image Viewer Queries

To create an Image Viewer Query, follow the instructions outlined above in Creating Queries. Set the Display As option to Image Viewer.

When displaying a record on an Image Viewer Xalt needs to know the Large Binary image or the URL that contains the image the record represents. Add the property to the Query's View Properties, setting the Usage to Image/Video. If you are using a String property that contains the URL, the Domain Object Property you add must have a Class of URL in order for the Image/Video usage to be available. You can also add one or more properties with a Usage of Identifier. Identifier properties will be concatenated together using a comma as a separator to identify a record on the map. If no Identifier property exits then the Data Object's short title will be used.

Multiple Queries of different types

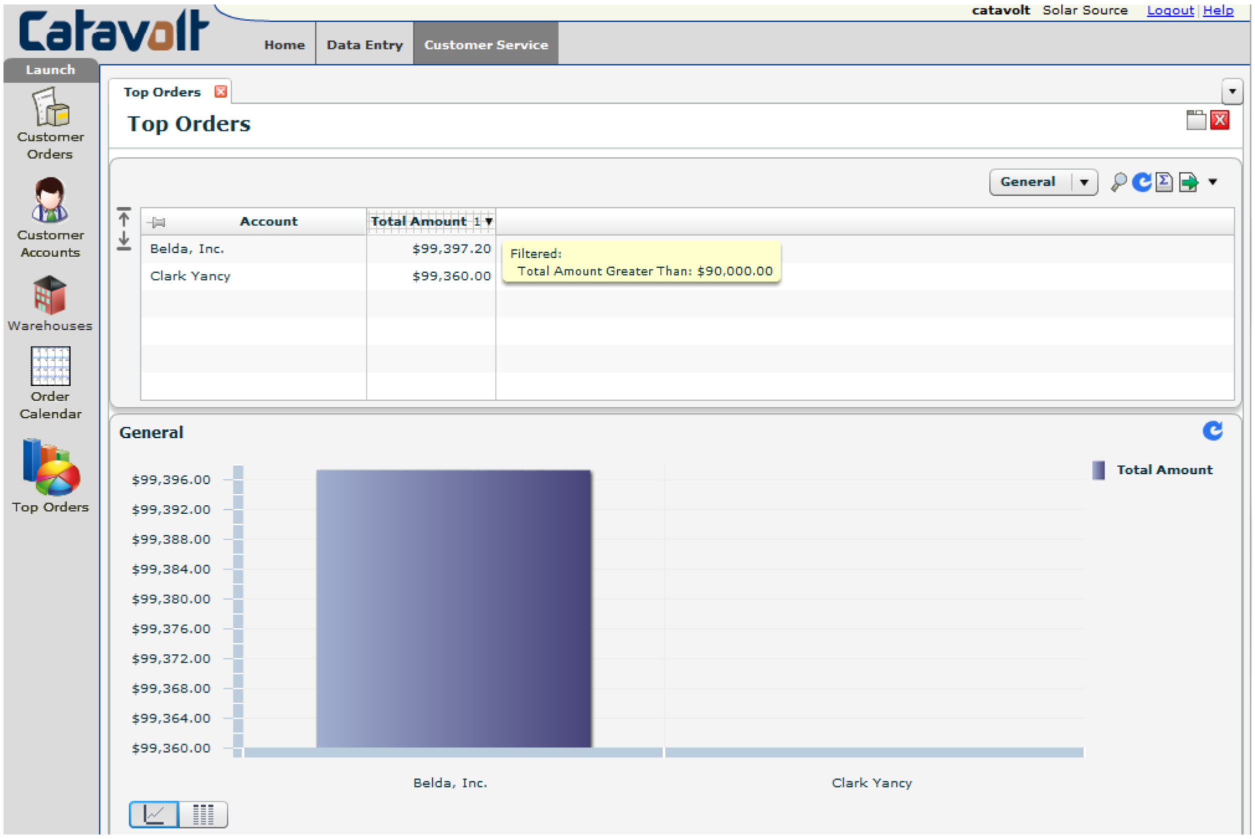
If you have defined multiple Queries of different types in your Data Object (for example, a Table and a Graph), these views will automatically be displayed together when displaying the Data Object in a Rich client:



Account	Total Amount 1 ▾
Belda, Inc.	$99,397.20
Clark Yancy	$99,360.00
Aeroflex Airflyte	$86,604.75
Liddy Corporation	$86,502.48
Vinyl Technology, Inc.	$86,400.00

Figure 34: A single Data Object view showing a table and a graph query

One benefit of having these Queries linked is that changes in the table will be automatically reflected in the other Queries. This allows for more complex filtering than what can be done using graph filters.



Account	Total Amount 1 ▼
Belda, Inc.	$99,397.20
Clark Yancy	$99,360.00

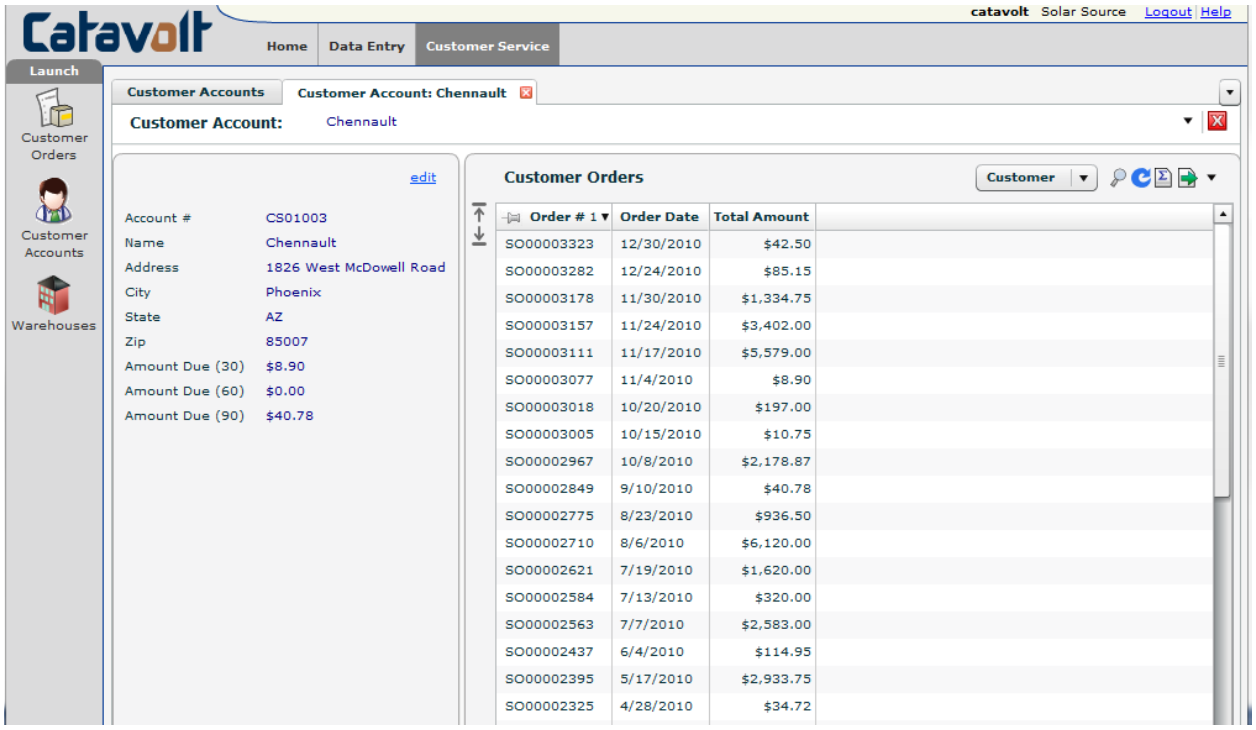
Filtered:
Total Amount Greater Than: $90,000.00

Figure 35: A single Data Object view showing a table and graph with table filtering in effect

Chapter 5: Data Object Details

Chapter Summary

A Detail represents a single record. A Detail allows you to more fully display the information about an individual record than what you can accomplish using a Query alone. Details can contain properties associated with the record being displayed as well as lists of related Data Objects. As with Queries, you are allowed to define multiple Details for a particular Data Object. Each Detail can provide a different view over the Data Object, so that you are not forced to attempt to put every possible field that may be of interest on a single Detail.



Customer Account: Chennault

Account #	CS01003
Name	Chennault
Address	1826 West McDowell Road
City	Phoenix
State	AZ
Zip	85007
Amount Due (30)	$8.90
Amount Due (60)	$0.00
Amount Due (90)	$40.78

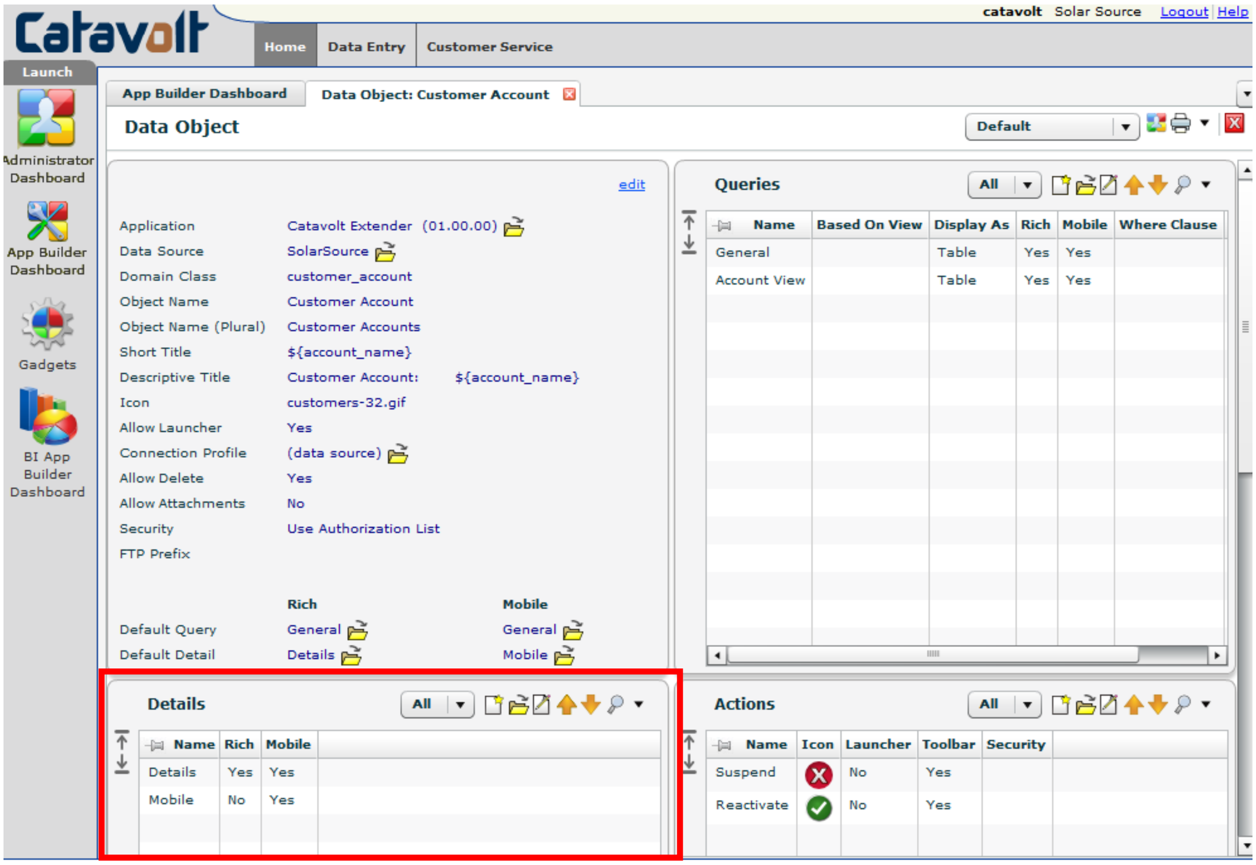
Customer Orders

Order # 1	Order Date	Total Amount
SO00003323	12/30/2010	$42.50
SO00003282	12/24/2010	$85.15
SO00003178	11/30/2010	$1,334.75
SO00003157	11/24/2010	$3,402.00
SO00003111	11/17/2010	$5,579.00
SO00003077	11/4/2010	$8.90
SO00003018	10/20/2010	$197.00
SO00003005	10/15/2010	$10.75
SO00002967	10/8/2010	$2,178.87
SO00002849	9/10/2010	$40.78
SO00002775	8/23/2010	$936.50
SO00002710	8/6/2010	$6,120.00
SO00002621	7/19/2010	$1,620.00
SO00002584	7/13/2010	$320.00
SO00002563	7/7/2010	$2,583.00
SO00002437	6/4/2010	$114.95
SO00002395	5/17/2010	$2,933.75
SO00002325	4/28/2010	$34.72

Figure 1: An example Data Object Detail in a web browser

Accessing Details

To access the Details defined for a particular Data Object, open the Data Object and look at the Default Detail. Multiple Details can be created for a Data Object. They will appear in the Detail dropdown list in the same order they appear in the Details section. To change the order, select a Detail and press the Move Up and Move Down toolbar buttons.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Short Title	${account_name}
Descriptive Title	Customer Account: ${account_name}
Icon	customers-32.gif
Allow Launcher	Yes
Connection Profile	(data source)
Allow Delete	Yes
Allow Attachments	No
Security	Use Authorization List
FTP Prefix	

	Rich	Mobile
Default Query	General	General
Default Detail	Details	Mobile

Queries

Name	Based On View	Display As	Rich	Mobile	Where Clause
General		Table	Yes	Yes	
Account View		Table	Yes	Yes	

Details

Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

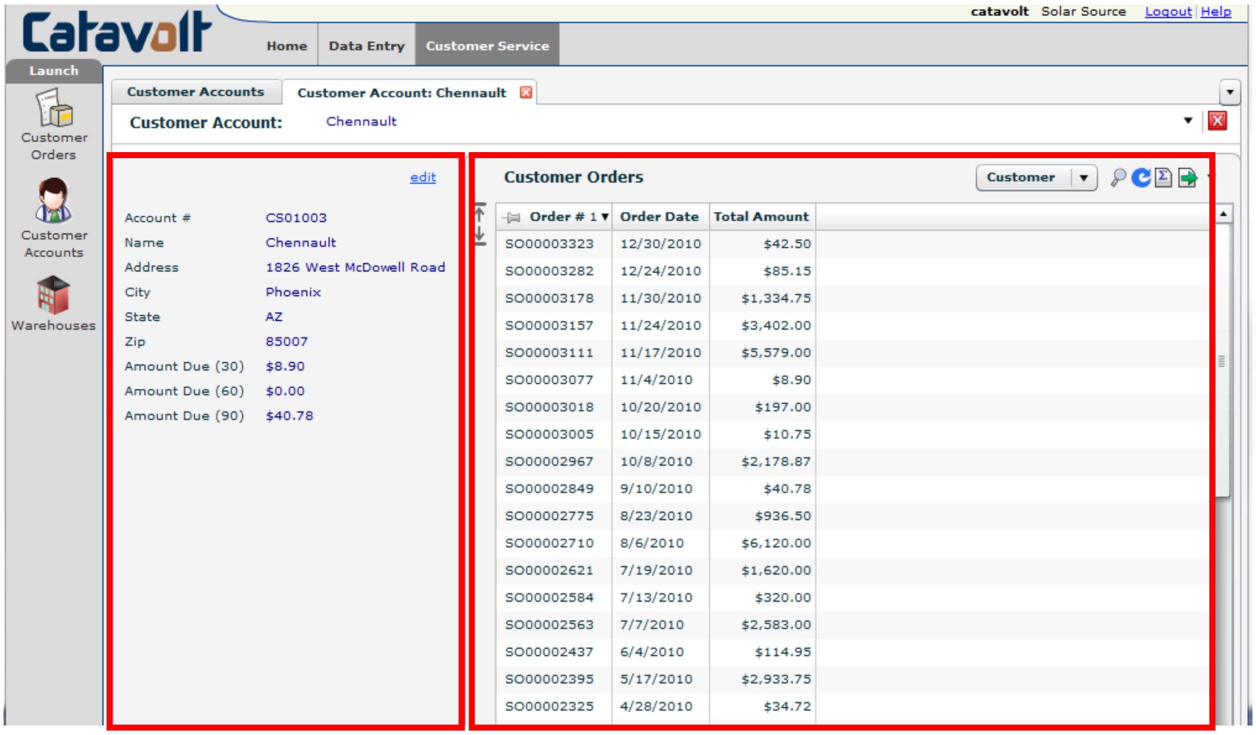
Actions

Name	Icon	Launcher	Toolbar	Security
Suspend	✗	No	Yes	
Reactivate	✓	No	Yes	

Figure 2: Data Object Default details view with the Details query section highlighted

Detail Components

When you display the definition for a Detail, there are 2 Details that you can select from. The Default Detail shows you all the information about the Detail. Data Object Details contains 2 components: Sections and Hidden Actions (a list of Actions to hide from the menu when this Detail is Active). A section roughly corresponds to a rectangular area. Notice in the screenshot below, there are 2 sections: a Property Section on top with a Query Section to the right:



Customer Account: Chennault

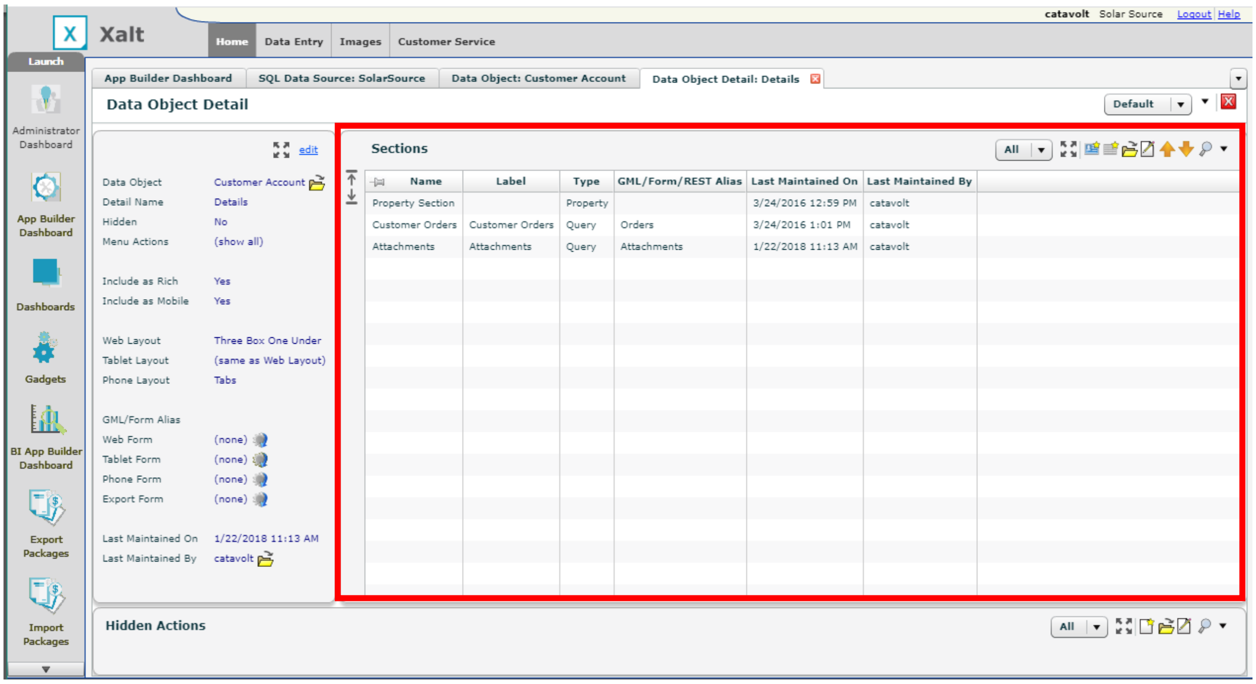
Account #	CS01003
Name	Chennault
Address	1826 West McDowell Road
City	Phoenix
State	AZ
Zip	85007
Amount Due (30)	$8.90
Amount Due (60)	$0.00
Amount Due (90)	$40.78

Customer Orders

Order # 1	Order Date	Total Amount
SO00003323	12/30/2010	$42.50
SO00003282	12/24/2010	$85.15
SO00003178	11/30/2010	$1,334.75
SO00003157	11/24/2010	$3,402.00
SO00003111	11/17/2010	$5,579.00
SO00003077	11/4/2010	$8.90
SO00003018	10/20/2010	$197.00
SO00003005	10/15/2010	$10.75
SO00002967	10/8/2010	$2,178.87
SO00002849	9/10/2010	$40.78
SO00002775	8/23/2010	$936.50
SO00002710	8/6/2010	$6,120.00
SO00002621	7/19/2010	$1,620.00
SO00002584	7/13/2010	$320.00
SO00002563	7/7/2010	$2,583.00
SO00002437	6/4/2010	$114.95
SO00002395	5/17/2010	$2,933.75
SO00002325	4/28/2010	$34.72

Figure 3: An example Data Object Details in a web browser with two sections highlighted

Sections are listed when defining a Detail. They will appear in the same order they appear in the Sections list. To change the order, select a Section and press the Move Up and Move Down toolbar buttons.



Data Object Detail

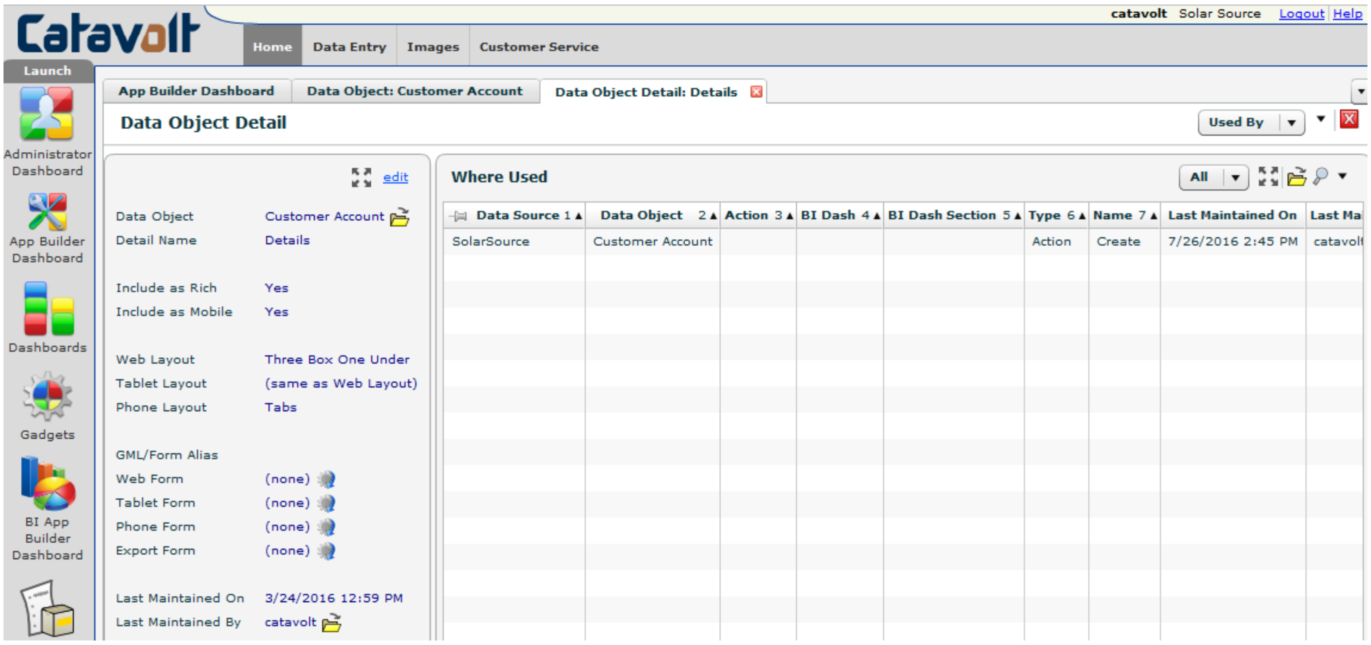
Data Object	Customer Account
Detail Name	Details
Hidden	No
Menu Actions	(show all)
Include as Rich	Yes
Include as Mobile	Yes
Web Layout	Three Box One Under
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	1/22/2018 11:13 AM
Last Maintained By	catavolt

Sections

Name	Label	Type	GML/Form/REST Alias	Last Maintained On	Last Maintained By
Property Section		Property		3/24/2016 12:59 PM	catavolt
Customer Orders	Customer Orders	Query	Orders	3/24/2016 1:01 PM	catavolt
Attachments	Attachments	Query	Attachments	1/22/2018 11:13 AM	catavolt

Figure 4: A Data Object Detail details view with the Sections query section highlighted

The Used By Detail shows a Where Used list of objects that are currently using this Detail.



Data Object Detail

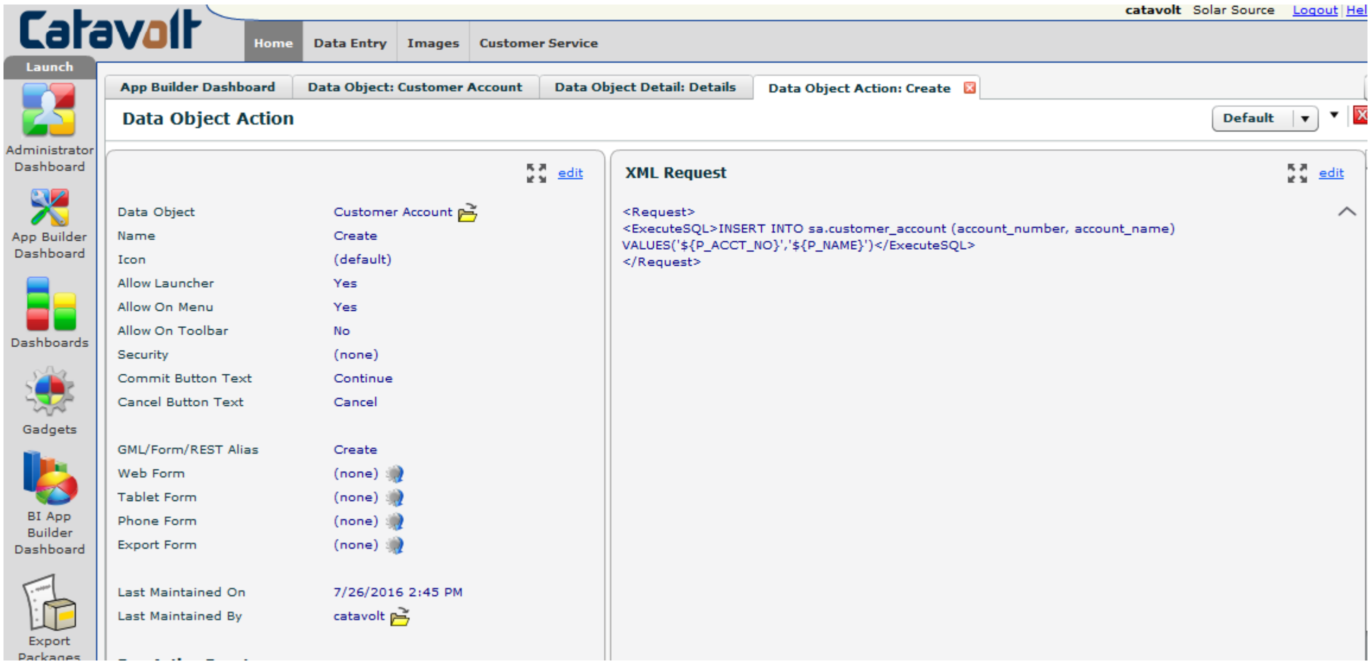
Data Object	Customer Account
Detail Name	Details
Include as Rich	Yes
Include as Mobile	Yes
Web Layout	Three Box One Under
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	3/24/2016 12:59 PM
Last Maintained By	catavolt

Where Used

Data Source 1	Data Object 2	Action 3	BI Dash 4	BI Dash Section 5	Type 6	Name 7	Last Maintained On	Last Ma
SolarSource	Customer Account				Action	Create	7/26/2016 2:45 PM	catavolt

Figure 5: Detail 'Used By' details

Opening a Where Used record will take you directly to the object using the Detail:



Data Object Action

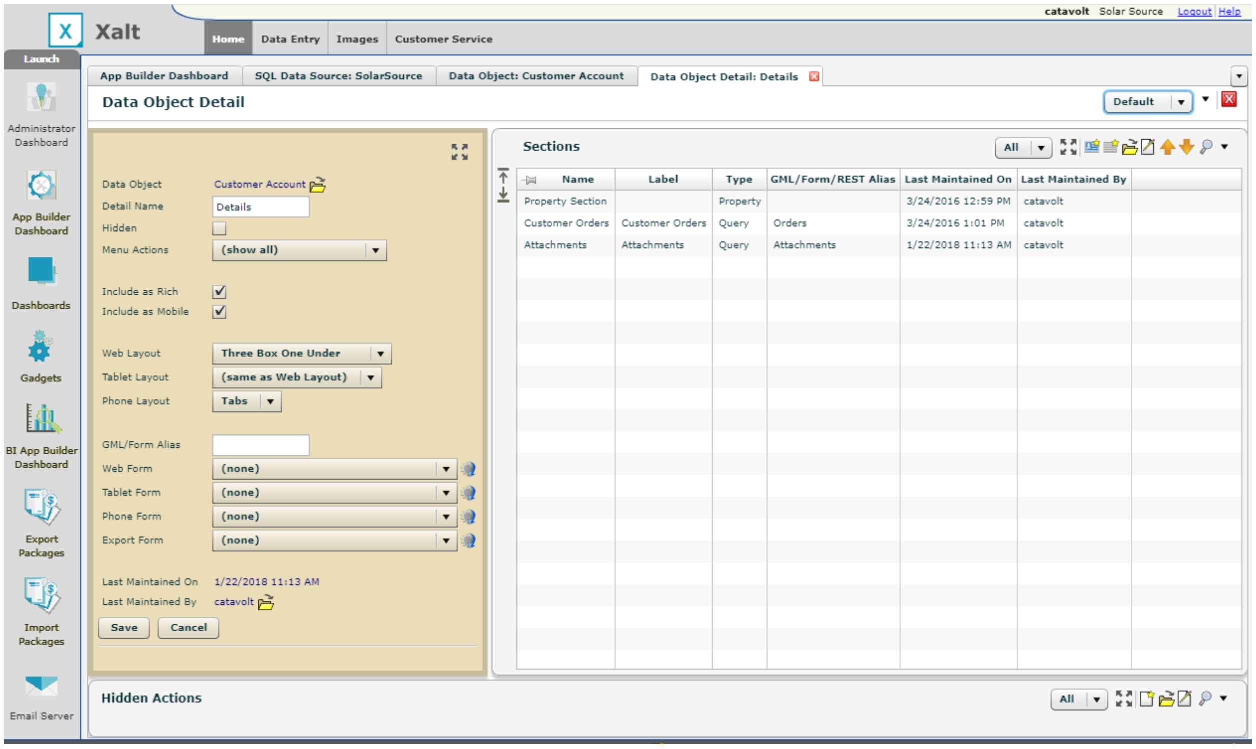
Data Object	Customer Account
Name	Create
Icon	(default)
Allow Launcher	Yes
Allow On Menu	Yes
Allow On Toolbar	No
Security	(none)
Commit Button Text	Continue
Cancel Button Text	Cancel
GML/Form/REST Alias	Create
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	7/26/2016 2:45 PM
Last Maintained By	catavolt

XML Request

```
<Request>
<ExecuteSQL>INSERT INTO sa.customer_account (account_number, account_name)
VALUES('${P_ACCT_NO}','${P_NAME}')</ExecuteSQL>
</Request>
```

Figure 6: Detail Where Used

Creating Details



Data Object Detail

Data Object	Customer Account
Detail Name	Details
Hidden	□
Menu Actions	(show all)
Include as Rich	☑
Include as Mobile	☑
Web Layout	Three Box One Under
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	1/22/2018 11:13 AM
Last Maintained By	catavolt

Sections

Name	Label	Type	GML/Form/REST Alias	Last Maintained On	Last Maintained By
Property Section		Property		3/24/2016 12:59 PM	catavolt
Customer Orders	Customer Orders	Query	Orders	3/24/2016 1:01 PM	catavolt
Attachments	Attachments	Query	Attachments	1/22/2018 11:13 AM	catavolt

Hidden Actions

Figure 7: Query Object Detail in edit mode

When creating a Data Object Detail, you must specify the Detail Name. This value is used to uniquely identify the Detail and is used in the details drop down at runtime.

Hidden specifies whether this Detail will appear in the runtime Detail dropdown list. If Hidden is selected, this Detail will not appear, nor will it appear in the Data Object's Default Rich/Mobile Detail list. The Detail can still be explicitly selected in other places (e.g. in a Completion Action Rich/Mobile Detail).

Menu Actions – specifies how to display the menu when this Detail is active. The list of available values are:

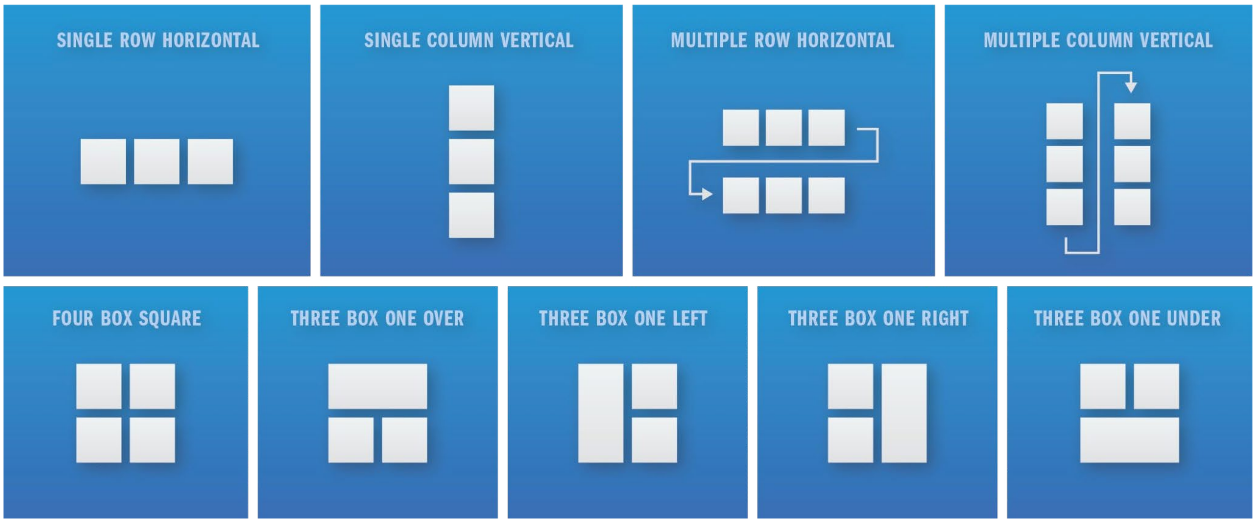
- (show all) – Show all Actions on the menu. This is the default
- (hide all) – Hide all Actions (standard as well as custom) from the menu. You can choose this option to quickly hide the entire menu without having to add every action to the Hidden Actions List.
- Read-only Standard Actions -- Hide all custom Actions as well any standard Actions that maintain data (Create, Update, Delete, and Attach File (for Infor XA System-Link Data Objects)). You can choose this option to quickly make the menu "read-only" without having to add every action to the Hidden Actions List.
- Use Hidden Actions List – Specify a list of Actions to hide from the menu when this Detail is active.

Note that you can still call an action explicitly in a Custom Form or GML screen even if it is hidden from the menu.

Include as Rich specifies that this detail is available to use on rich clients, such as the Web and Gadgets.

Include as Mobile specifies that this detail is available to use on mobile clients.

Web Layout specifies how you want to arrange the different sections when viewed by a Web Browser. The following options are available:



- Single Row Horizontal – All sections are arrange side-by-side in a single row
- Single Column Vertical – All sections are arrange on top of each other in a single column
- Multiple Row Horizontal – Sections are arranged side-by-side. Sections fill up a row based on the window size and then wrap to the next row as needed. As the window is resized the sections adjust and wrap. This is similar to word wrapping in word processing applications.
- Multiple Column Vertical – Sections are arranged on top of each other. Sections fill up a column based on the window size and then wrap to the top of the next column. As the window is resized the sections adjust and wrap.
- Four Box Square – Up to 4 sections are equally sized and displayed in a 2x2 area. Any sections beyond the fourth are added underneath using the "Single Column Vertical" style.
- Three Box One Over – Up to 3 sections are placed, the first on row one and the other two side-by-side on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Left – Up to 3 sections are placed, the first on column one and the other two on top of each other on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.

- Three Box One Right – Up to 3 sections are placed, the first two on top of each other on column one and the third on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Under – Up to 3 sections are placed, the first two side-by-side on row one and the third on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Tabs – Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

Tablet Layout specifies how you want to arrange the different sections when viewed on a Tablet-size mobile device. The following options are available:

- (same as web layout) – Use the same layout as the rich client, if it is available on mobile. If the web layout is not available on the mobile device, Single Column Vertical will be used.
- Single Column Vertical – All sections are arrange on top of each other in a single column
- Four Box Square – Up to 4 sections are equally sized and displayed in a 2x2 area. Any sections beyond the fourth are added underneath using the "Single Column Vertical" style.
- Three Box One Over – Up to 3 sections are placed, the first on row one and the other two side-by-side on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Left – Up to 3 sections are placed, the first on column one and the other two on top of each other on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Right – Up to 3 sections are placed, the first two on top of each other on column one and the third on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Under – Up to 3 sections are placed, the first two side-by-side on row one and the third on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Tabs - Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

Phone Layout specifies how you want to arrange the different sections when viewed on a Phone-size mobile device. The following options are available:

- Tabs - Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

GML/Form Alias allows you to specify an alias when using this Detail with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Web Form specifies which custom form will be used to display the data for this detail when on a Web Browser.

Tablet Form specifies which custom form will be used to display the data for this detail when on a tablet device.

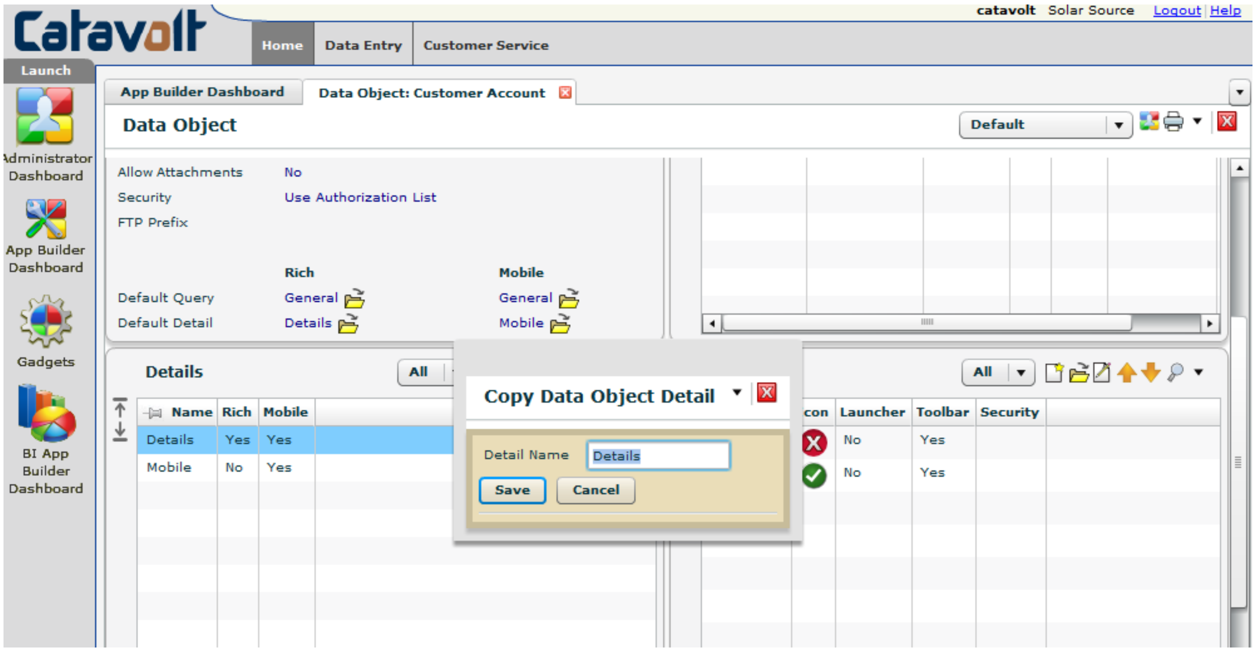
Phone Form specifies which custom form will be used to display the data for this detail when on a phone device.

Export Form specifies which custom form will be used to display the data for this detail when choosing the Export Form to PDF menu option. Selecting (none) will remove the Export Form to PDF menu option from this detail.

You can also view, upload, and remove custom forms from a dedicated launcher. See Chapter 13: Image/Asset Management for more information.

Copying Details

You may have instances where you need to make a copy of an existing Detail to show a different set of properties. You can select the Copy menu option to accomplish this. When Copying a Detail, you will be prompted to supply the new Detail Name. An exact copy of this Detail along with all of its components will be made.



Data Object

Allow Attachments	No	
Security	Use Authorization List	
FTP Prefix		
	Rich	Mobile
Default Query	General	General
Default Detail	Details	Mobile

Details

Name	Rich	Mobile
Details	Yes	Yes
Mobile	No	Yes

Copy Data Object Detail

Detail Name: Details

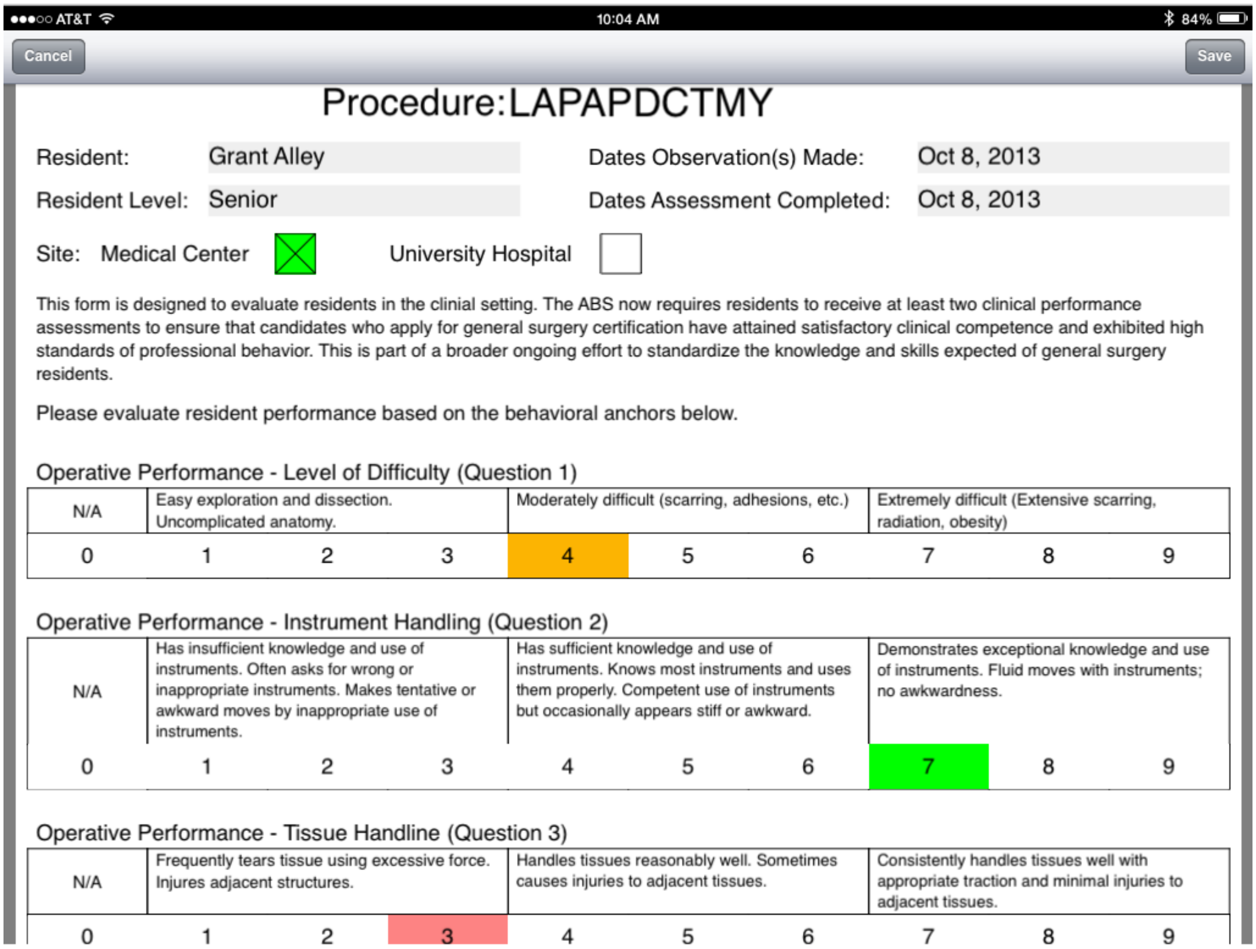
Save Cancel

con	Launcher	Toolbar	Security
✗	No	Yes	
✓	No	Yes	

Figure 8: The Data Object Detail copy prompt

Upload Custom Form

Xalt | Mobility has the capability to render Details as custom interactive forms. As you can see from the examples below, details can be displayed and edited using layouts and controls not available in the regular Xalt product.



Procedure:LAPAPDCTMY

Resident: Grant Alley

Resident Level: Senior

Dates Observation(s) Made: Oct 8, 2013

Dates Assessment Completed: Oct 8, 2013

Site: Medical Center ☒ University Hospital □

This form is designed to evaluate residents in the clinial setting. The ABS now requires residents to receive at least two clinical performance assessments to ensure that candidates who apply for general surgery certification have attained satisfactory clinical competence and exhibited high standards of professional behavior. This is part of a broader ongoing effort to standardize the knowledge and skills expected of general surgery residents.

Please evaluate resident performance based on the behavioral anchors below.

Operative Performance - Level of Difficulty (Question 1)

N/A	Easy exploration and dissection. Uncomplicated anatomy.			Moderately difficult (scarring, adhesions, etc.)			Extremely difficult (Extensive scarring, radiation, obesity)		
0	1	2	3	[✓ 4]	5	6	7	8	9

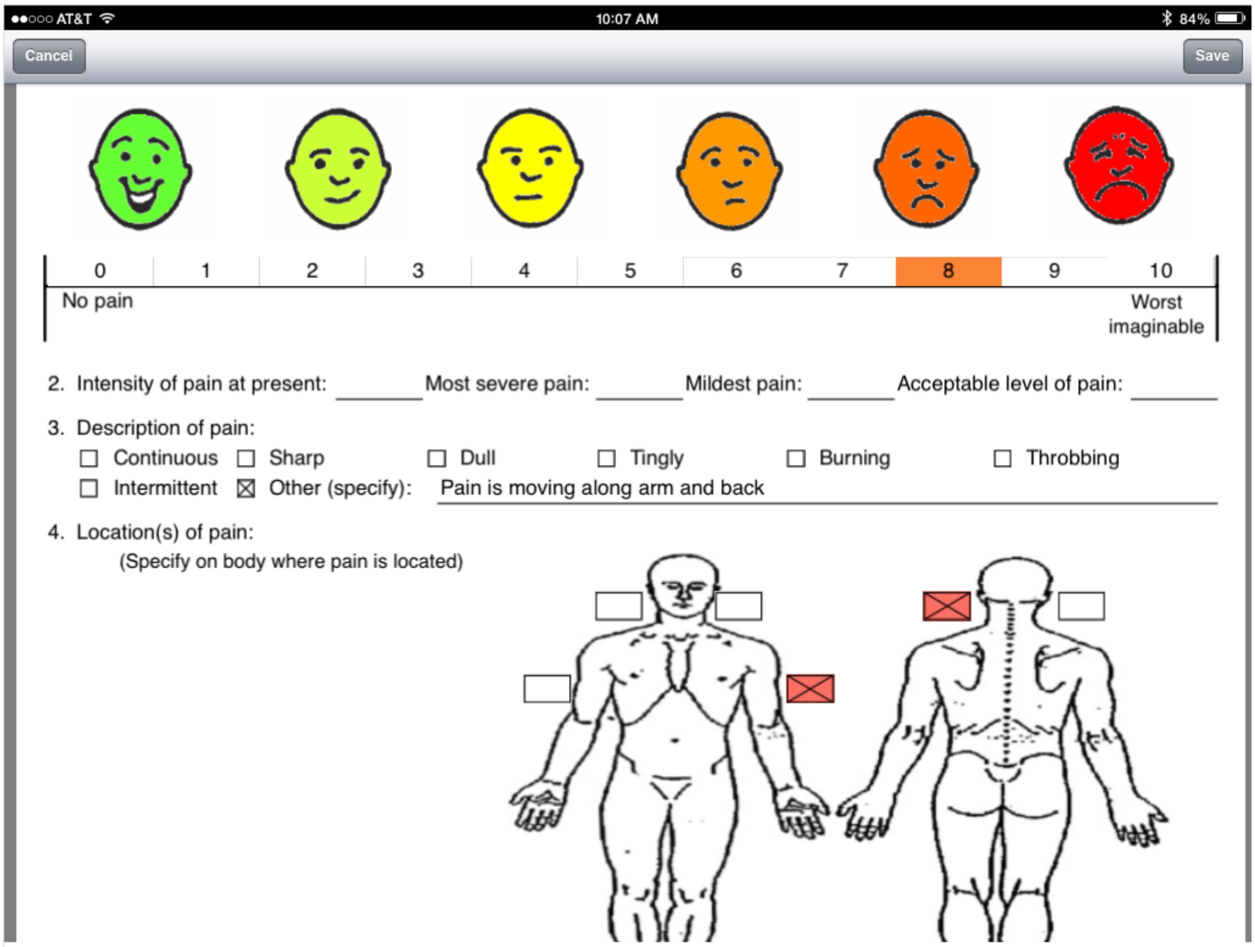
Operative Performance - Instrument Handling (Question 2)

N/A	Has insufficient knowledge and use of instruments. Often asks for wrong or inappropriate instruments. Makes tentative or awkward moves by inappropriate use of instruments.			Has sufficient knowledge and use of instruments. Knows most instruments and uses them properly. Competent use of instruments but occasionally appears stiff or awkward.			Demonstrates exceptional knowledge and use of instruments. Fluid moves with instruments; no awkwardness.		
0	1	2	3	4	5	6	[✓ 7]	8	9

Operative Performance - Tissue Handline (Question 3)

N/A	Frequently tears tissue using excessive force. Injures adjacent structures.			Handles tissues reasonably well. Sometimes causes injuries to adjacent tissues.			Consistently handles tissues well with appropriate traction and minimal injuries to adjacent tissues.		
0	1	2	[✓ 3]	4	5	6	7	8	9

Figure 9: An example of a Custom Form in Xalt



0	1	2	3	4	5	6	7	8	9	10
No pain										Worst imaginable

2. Intensity of pain at present: ________ Most severe pain: ________ Mildest pain: ________ Acceptable level of pain: ________

3. Description of pain:
☐ Continuous ☐ Sharp ☐ Dull ☐ Tingly ☐ Burning ☐ Throbbing
☐ Intermittent ☒ Other (specify): Pain is moving along arm and back

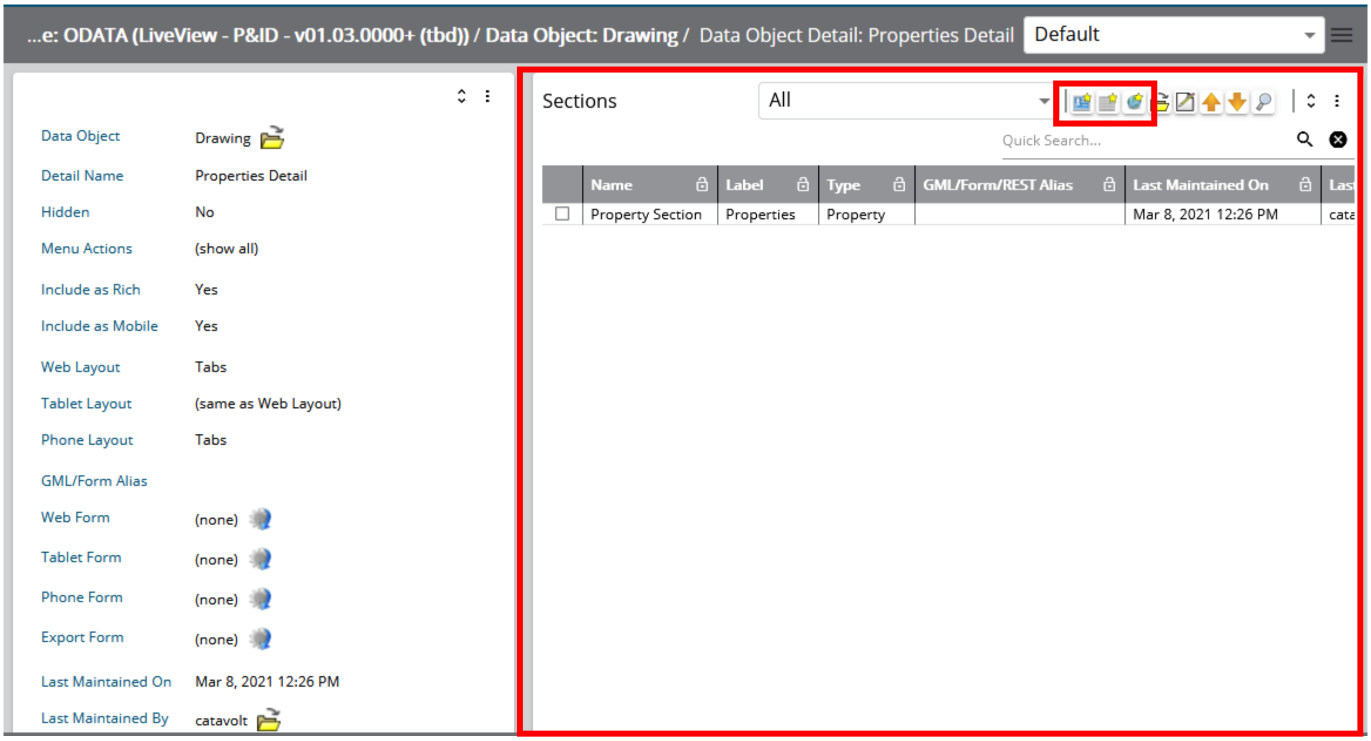
4. Location(s) of pain:
(Specify on body where pain is located)

Figure 10: An example of a Custom Form in Xalt

The Upload Custom Form menu option allows you to replace the existing Detail definition with an Xalt Custom Form. You can also view, upload, and remove custom forms from a dedicated launcher. See Chapter 13: Image/Asset Management for more information. Your Hexagon sales representative has more information on how to create and deploy Custom Forms.

Detail Sections

The Sections section shows a list of all sections defined for the specified details. There are three kinds of Sections: Property Sections, Query Sections, and Visualization Sections (Note: Visualization Sections are only available for ODATA Data Sources).



...e: ODATA (LiveView - P&ID - v01.03.0000+ (tbd)) / Data Object: Drawing / Data Object Detail: Properties Detail

Default

Data Object	Drawing
Detail Name	Properties Detail
Hidden	No
Menu Actions	(show all)
Include as Rich	Yes
Include as Mobile	Yes
Web Layout	Tabs
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	Mar 8, 2021 12:26 PM
Last Maintained By	catavolt

Sections All

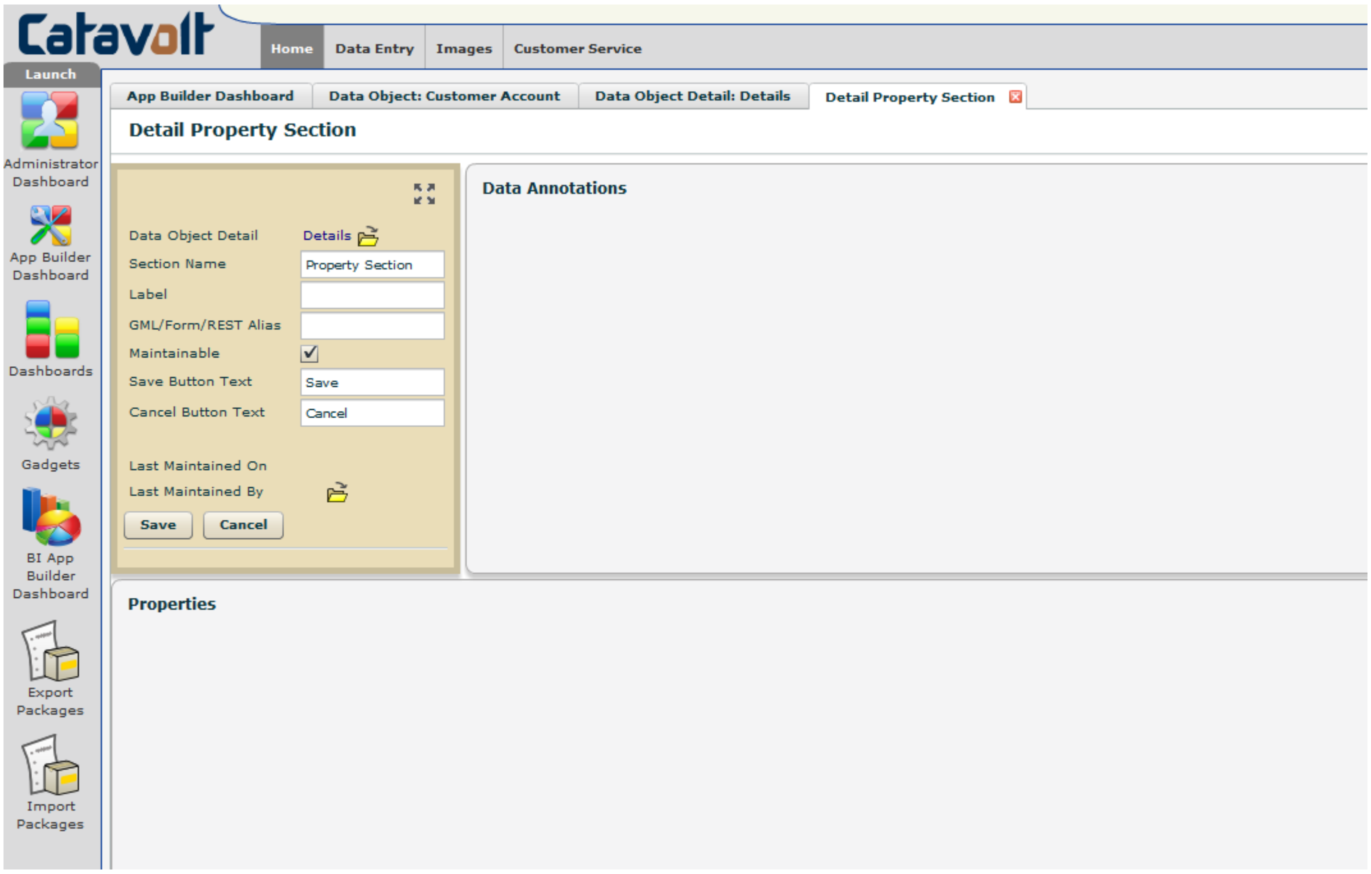
Quick Search...

	Name	Label	Type	GML/Form/REST Alias	Last Maintained On	Las
☐	Property Section	Properties	Property		Mar 8, 2021 12:26 PM	cata

Figure 11: Data Object Detail details view with the Sections query section highlighted along with the Add Property Section, Add Query Section, and Add Visualization Section toolbar actions

Property Section

Like its name implies, a Property Section contains Properties and Constants from the selected Data Object.



Detail Property Section

Data Object Detail	Details
Section Name	Property Section
Label	
GML/Form/REST Alias	
Maintainable	☑
Save Button Text	Save
Cancel Button Text	Cancel
Last Maintained On	
Last Maintained By	

Figure 12: Detail Property Section details view

When creating a Detail Property Section, you must specify the Section Name. This value is used to uniquely identify the Property Section and is not presented to the end user.

Label specifies an identifying description of the section as displayed to the end user in the top left corner. Leaving this value empty will cause no label to be displayed for the section.

GML/Form Alias allows you to specify an alias when using this Section with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

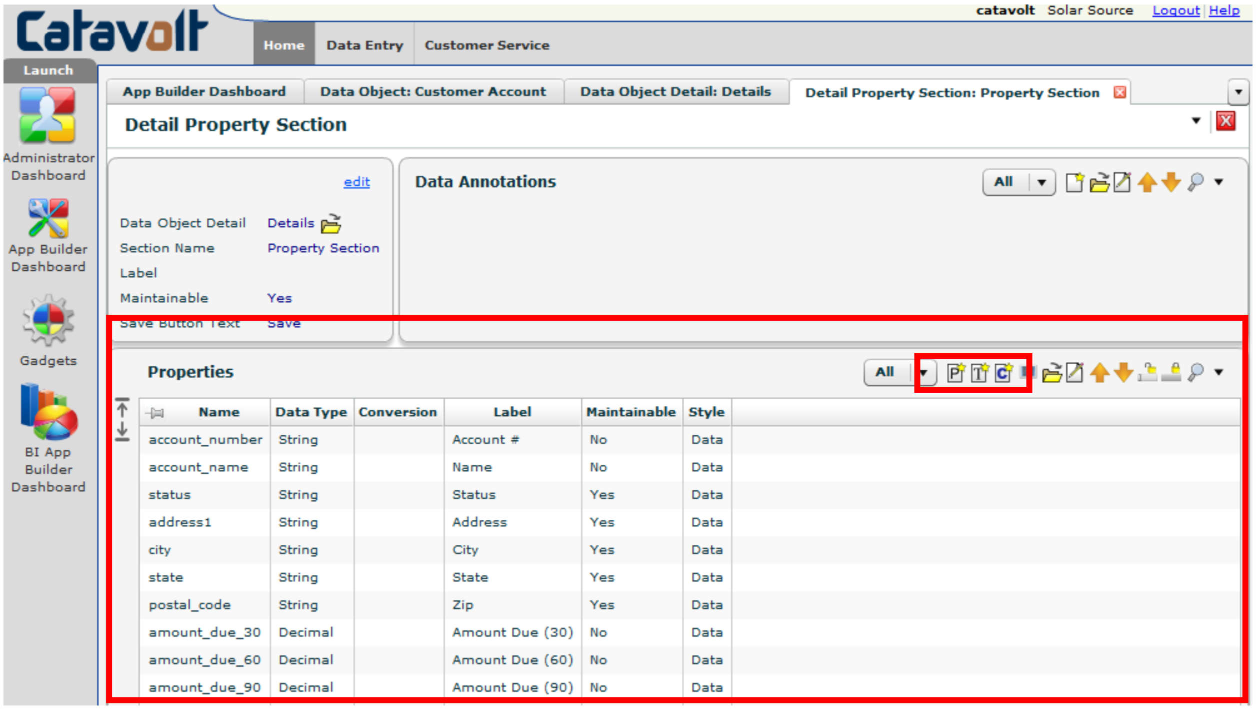
Maintainable specifies whether the section is to remain read-only or can be can maintained by the end user. Selecting this checkbox will add an Edit hotlink in the top right corner of the Property Section.

Save Button Text specifies the label that the Save button will show when the property section is being edited. The default value is Save.

Cancel Button Text specifies the label that the Cancel button will show when the property section is being edited. The default value is Cancel.

Property Section Properties

There are 4 types of Properties that can be added to a Property Section: Data Object Properties, Text Properties, Constant Properties, and Column Breaks. Properties will appear in a Property Section in the same order they appear in the Properties section. To change the order, select a Property and press the Move Up and Move Down toolbar buttons.



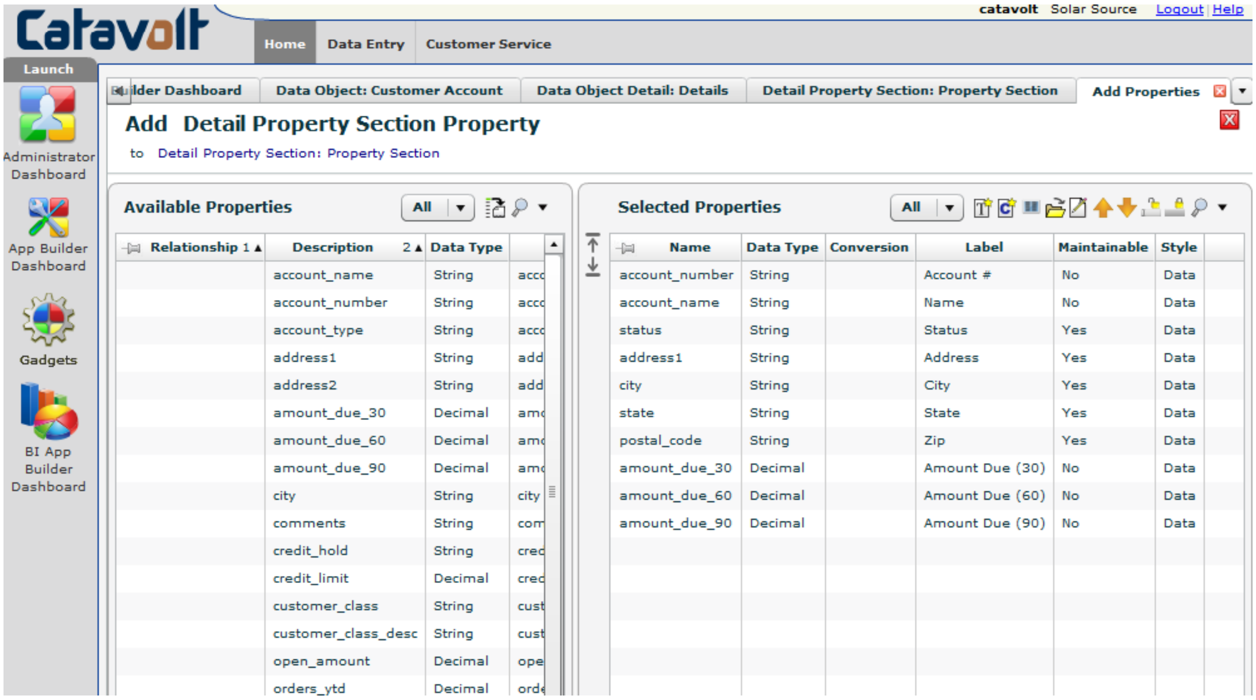
Data Object Detail	Details
Section Name	Property Section
Label	
Maintainable	Yes
Save Button Text	Save

Name	Data Type	Conversion	Label	Maintainable	Style
account_number	String		Account #	No	Data
account_name	String		Name	No	Data
status	String		Status	Yes	Data
address1	String		Address	Yes	Data
city	String		City	Yes	Data
state	String		State	Yes	Data
postal_code	String		Zip	Yes	Data
amount_due_30	Decimal		Amount Due (30)	No	Data
amount_due_60	Decimal		Amount Due (60)	No	Data
amount_due_90	Decimal		Amount Due (90)	No	Data

Figure 13: Data Object Detail details view with the Properties query section highlighted along with the Add Property, Add Text Property, Add Column toolbar actions

Data Object Properties

When adding Data Object Properties, you will be presented with two lists. The Available Properties list shows all properties for the Data Object (including properties from related foreign keys and Additional Query Scopes). The Selected Properties list shows the properties that will be displayed on the Property Section



Add Detail Property Section Property

to Detail Property Section: Property Section

Available Properties

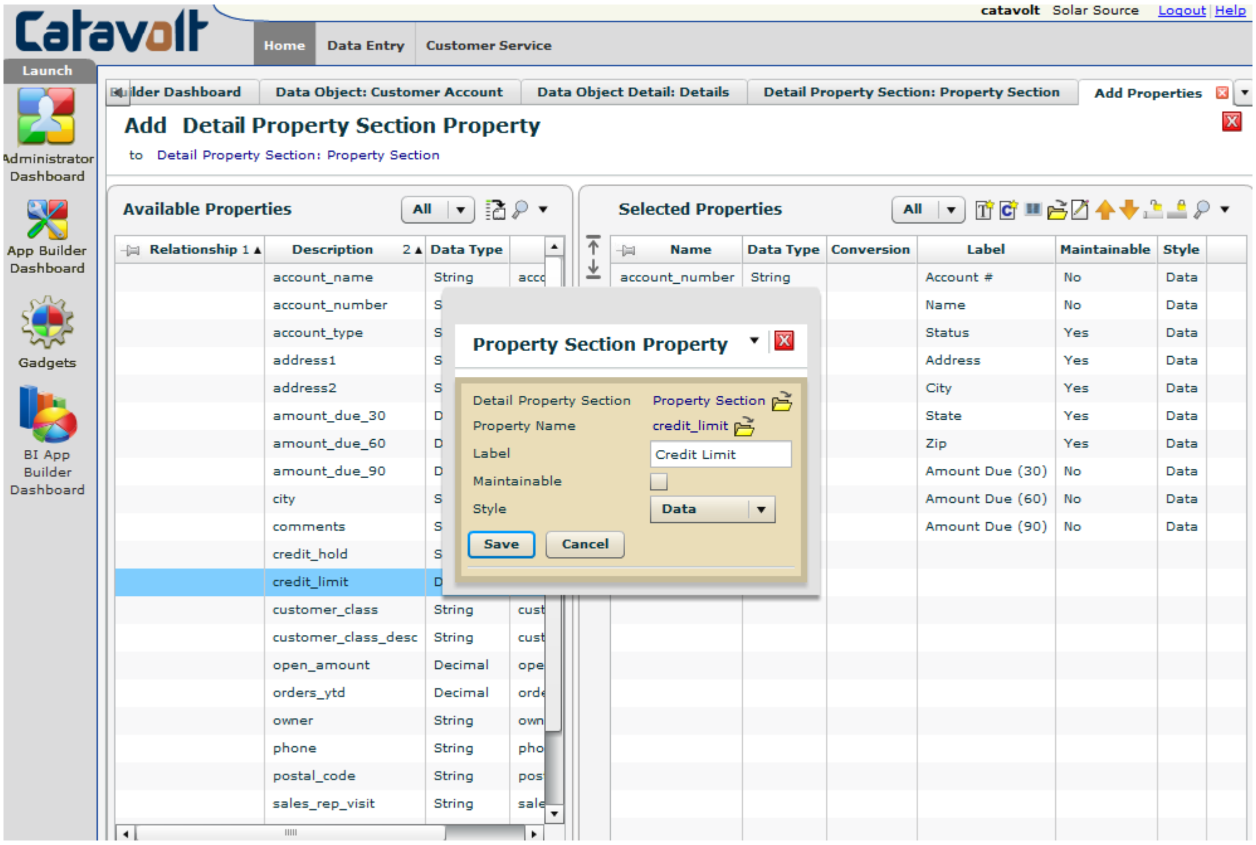
Relationship 1 ▲	Description 2 ▲	Data Type	
	account_name	String	acco
	account_number	String	acco
	account_type	String	acco
	address1	String	add
	address2	String	add
	amount_due_30	Decimal	amo
	amount_due_60	Decimal	amo
	amount_due_90	Decimal	amo
	city	String	city
	comments	String	com
	credit_hold	String	cred
	credit_limit	Decimal	cred
	customer_class	String	cust
	customer_class_desc	String	cust
	open_amount	Decimal	ope
	orders_ytd	Decimal	orde

Selected Properties

Name	Data Type	Conversion	Label	Maintainable	Style
account_number	String		Account #	No	Data
account_name	String		Name	No	Data
status	String		Status	Yes	Data
address1	String		Address	Yes	Data
city	String		City	Yes	Data
state	String		State	Yes	Data
postal_code	String		Zip	Yes	Data
amount_due_30	Decimal		Amount Due (30)	No	Data
amount_due_60	Decimal		Amount Due (60)	No	Data
amount_due_90	Decimal		Amount Due (90)	No	Data

Figure 14: The Add Detail Property Section Property view

You can select a single or multiple properties and press the Add Properties button to add them to the Property Section. If you select a single property, you will be presented with the following dialog:



Add Detail Property Section Property

to Detail Property Section: Property Section

Available Properties

Relationship 1 ▲	Description 2 ▲	Data Type	
	account_name	String	acco
	account_number	S	
	account_type	S	
	address1	S	
	address2	S	
	amount_due_30	D	
	amount_due_60	D	
	amount_due_90	D	
	city	S	
	comments	S	
	credit_hold	S	
	credit_limit	D	
	customer_class	String	cust
	customer_class_desc	String	cust
	open_amount	Decimal	ope
	orders_ytd	Decimal	orde
	owner	String	own
	phone	String	pho
	postal_code	String	pos
	sales_rep_visit	String	sale

Selected Properties

Name	Data Type	Conversion	Label	Maintainable	Style
account_number	String		Account #	No	Data
			Name	No	Data
			Status	Yes	Data
			Address	Yes	Data
			City	Yes	Data
			State	Yes	Data
			Zip	Yes	Data
			Amount Due (30)	No	Data
			Amount Due (60)	No	Data
			Amount Due (90)	No	Data

Property Section Property

Detail Property Section: Property Section
Property Name: credit_limit
Label: Credit Limit
Maintainable: □
Style: Data

Save Cancel

Figure 15: The add Property Section Property prompt for a single selection

Label specifies the description that will be displayed for the property. Xalt defaults this value to the Property's Description as specified in the Defined Properties section of the Data Object.

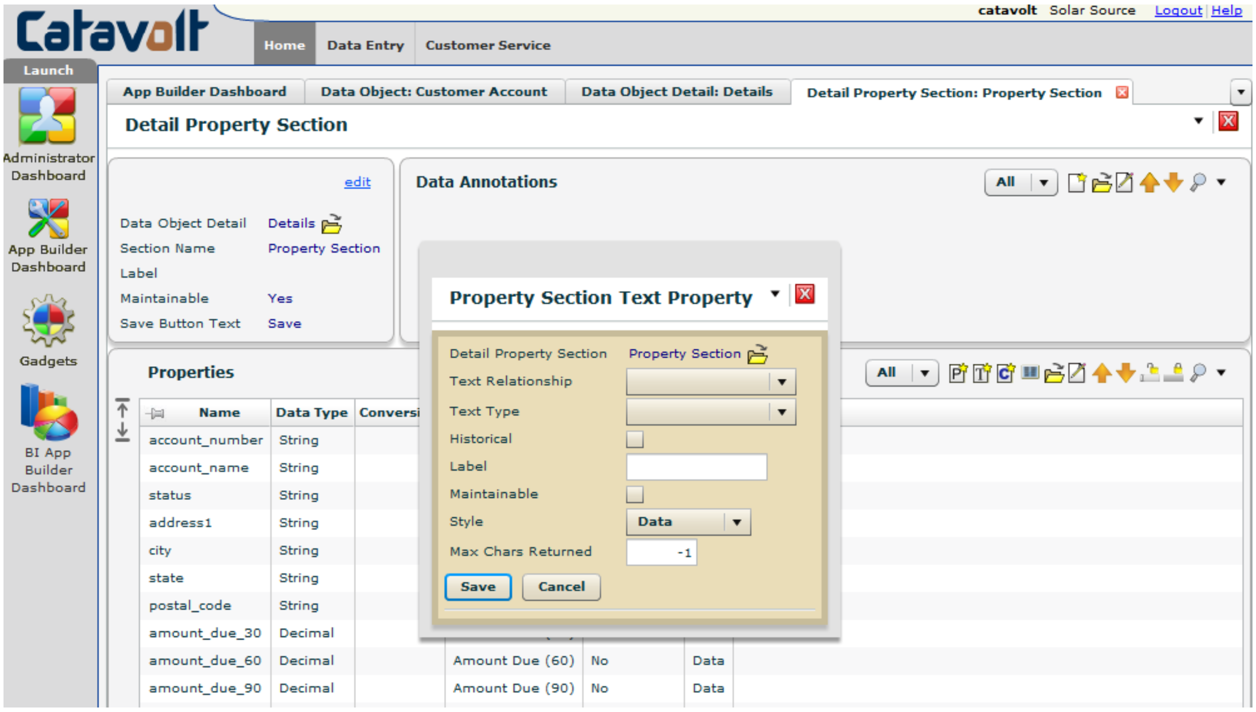
Maintainable specifies whether this property is read-only or can be changed by the end user. Xalt loads this value from the Property as specified in the Defined Properties section of the Data Object. Note that all references to a Property share the maintainable flag. Setting the value here will cause the value to change in all other Property sections that contain this property.

Style specifies the style that is used for displaying this value. Data presents it in a normal font, while Heading 1 through Heading 4 will display the value in bold with different font sizes (Heading 1 being the largest font). You may also specify Image/Video for URL and Large Binary properties.

You can get to the same dialog after adding the property to a Property Section by selecting it in the Selected Properties list and pressing the Edit button.

Text Properties

Some Data Sources (such as IDF System-Link) support the concept of separate Text Objects that are attached to Data Objects. For Data Sources that support this concept, you can add a Text Property. A Text Property is a multi-line entry field used to enter and display blocks of text.



Detail Property Section

edit

Data Object Detail	Details
Section Name	Property Section
Label	
Maintainable	Yes
Save Button Text	Save

Property Section Text Property

Detail Property Section	Property Section
Text Relationship	
Text Type	
Historical	□
Label	
Maintainable	□
Style	Data
Max Chars Returned	-1

Save Cancel

Name	Data Type	Conversi			
account_number	String				
account_name	String				
status	String				
address1	String				
city	String				
state	String				
postal_code	String				
amount_due_30	Decimal				
amount_due_60	Decimal		Amount Due (60)	No	Data
amount_due_90	Decimal		Amount Due (90)	No	Data

Figure 16: The add Property Section Text Property prompt

Text Relationship specifies which particular Text Object you wish to display for the given Data Object. If the Data Object does not contain any related Text Objects, (none available) will appear.

Text Type specifies which type will be displayed for the selected Text Relationship. If the Text Relationship does not contain multiple types, (none available) will appear.

Historical specifies whether the Text Object is a Historical Text Object (append allowed only) or a regular Text Object (full maintenance allowed).

Label specifies the description that will be displayed for the property. Xalt defaults this value to the Property's Description as specified in the Defined Properties section of the Data Object.

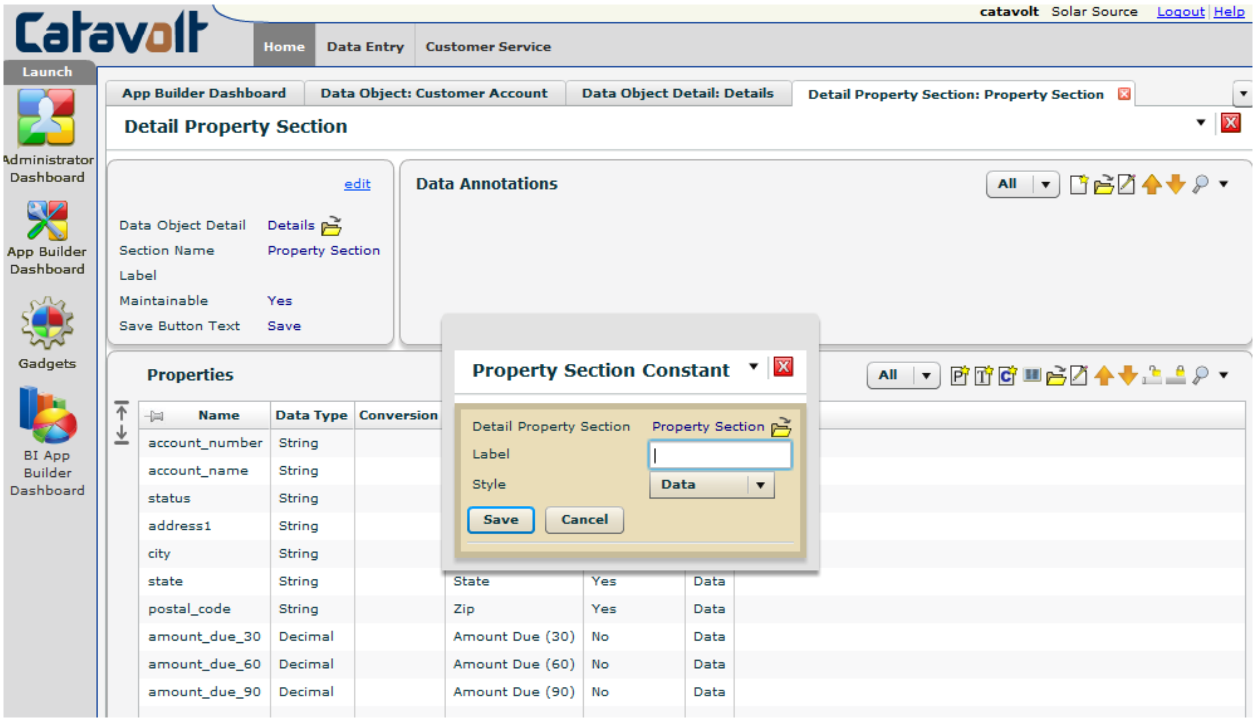
Maintainable specifies whether this Text Object is read-only or can be changed by the end user.

Style specifies the font that is used for displaying this value. Data presents it in a normal font, while Heading 1 through Heading 4 will display the value in bold with different font sizes (Heading 1 being the largest font).

Max Chars Returned specifies how much of the Text Object should be queried. Xalt defaults this value to -1, which will return the entire Text Object.

Constant Properties

Like their name implies, Constant Properties are text constants that can be used to further group data as it appears in Property Sections.



Detail Property Section

edit

Data Object Detail	Details
Section Name	Property Section
Label	
Maintainable	Yes
Save Button Text	Save

Property Section Constant

Detail Property Section	Property Section
Label	
Style	Data

Save Cancel

Name	Data Type	Conversion			
account_number	String				
account_name	String				
status	String				
address1	String				
city	String				
state	String		State	Yes	Data
postal_code	String		Zip	Yes	Data
amount_due_30	Decimal		Amount Due (30)	No	Data
amount_due_60	Decimal		Amount Due (60)	No	Data
amount_due_90	Decimal		Amount Due (90)	No	Data

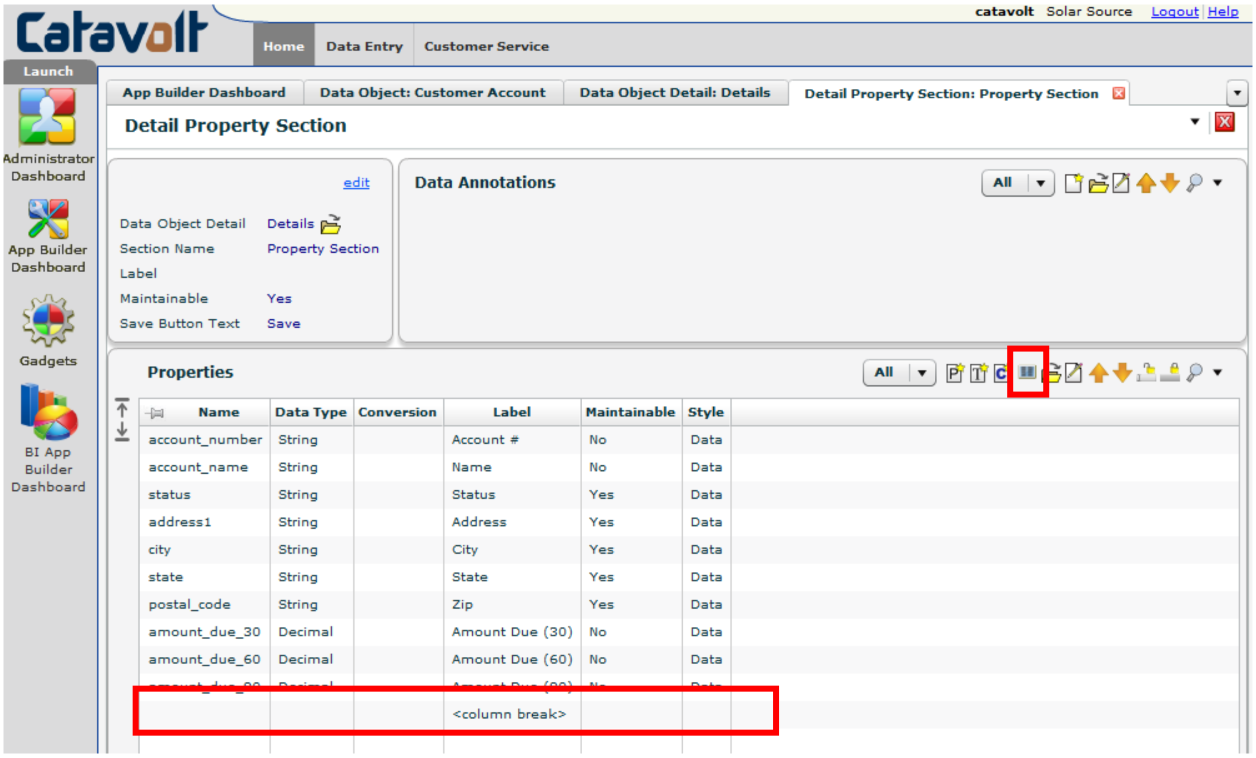
Figure 17: The add Property Section Constant prompt

Label specifies the description that will be displayed in the section. You can leave this value empty in order to add a blank line to the section.

Style specifies the font that is used for displaying this value. Data presents it in a normal font, while Heading 1 through Heading 4 will display the value in bold with different font sizes (Heading 1 being the largest font).

Column Breaks

Column Breaks allow you to display data across multiple columns within a Property Section.



Data Object Detail	Details
Section Name	Property Section
Label	
Maintainable	Yes
Save Button Text	Save

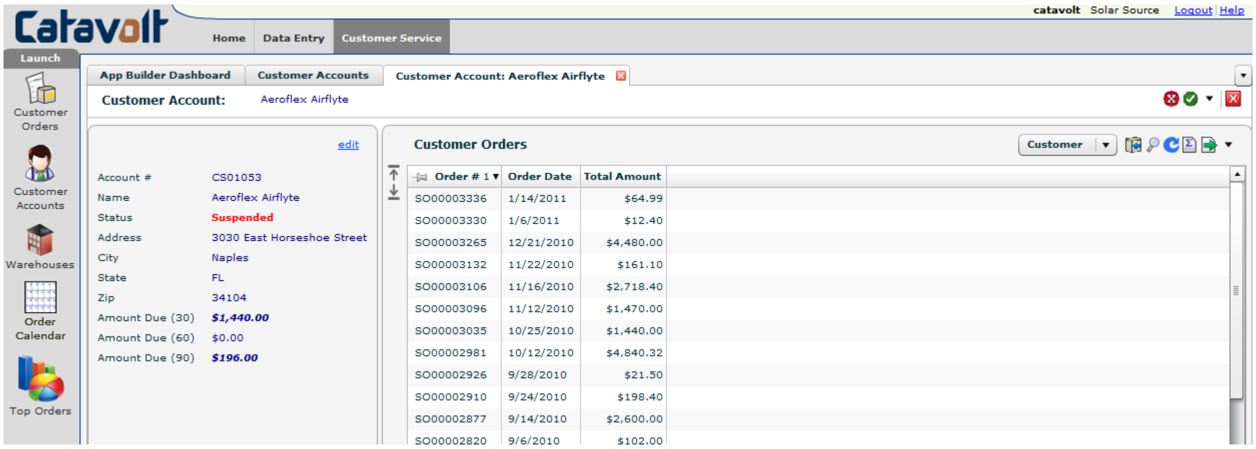
Name	Data Type	Conversion	Label	Maintainable	Style
account_number	String		Account #	No	Data
account_name	String		Name	No	Data
status	String		Status	Yes	Data
address1	String		Address	Yes	Data
city	String		City	Yes	Data
state	String		State	Yes	Data
postal_code	String		Zip	Yes	Data
amount_due_30	Decimal		Amount Due (30)	No	Data
amount_due_60	Decimal		Amount Due (60)	No	Data
amount_due_90	Decimal		Amount Due (90)	No	Data
			<column break>		

Figure 18: The Add Column action will add a Column Break indicator to the list

Selecting the Add Column menu action will add a <column break> indicator to the list of Properties. Everything above the <column break> indicator will be displayed in Column 1, while everything below the indicator will be displayed in Column 2. Multiple column breaks can be added to create more columns. In order for the data to line up correctly, you will need to ensure that columns to the left contain the same number of fields or more than columns to the right.

Data Annotations

Data Annotations for details are very similar to Data Annotations for queries in that they allow you to highlight Detail data using colors, fonts, images and alternate text. Data Annotations allow you to add background colors, foreground colors, bold lettering, italic lettering and override text based on the data being displayed. The end user sees the annotations in their Property Sections.



Customer Account: Aeroflex Airflyte

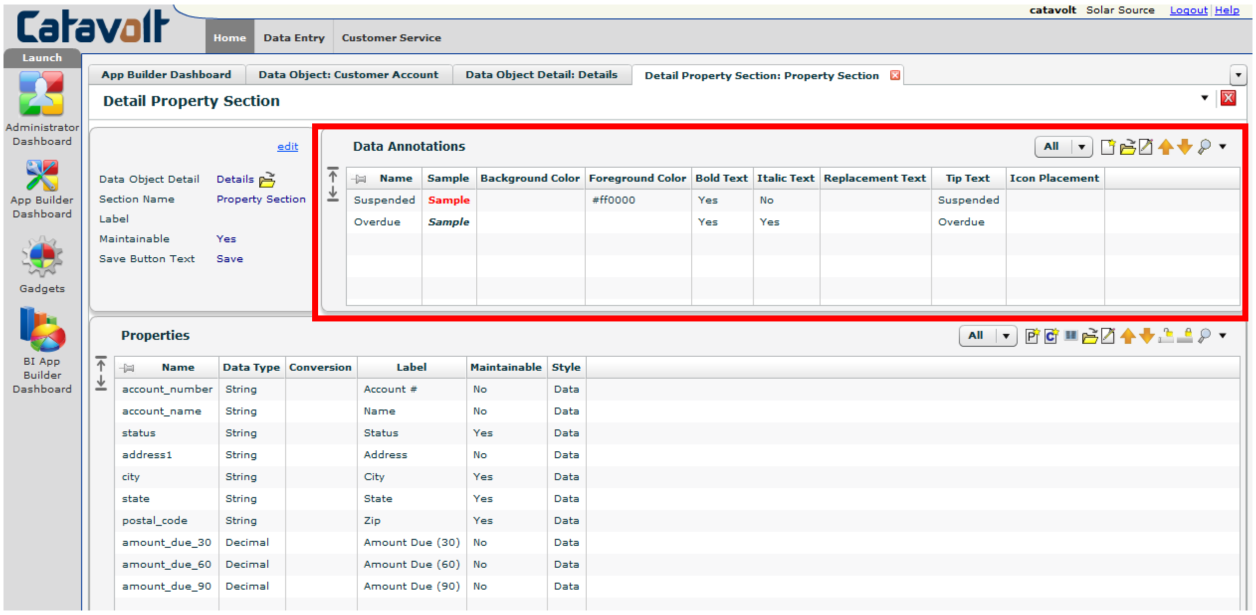
Account #	CS01053
Name	Aeroflex Airflyte
Status	**Suspended**
Address	3030 East Horseshoe Street
City	Naples
State	FL
Zip	34104
Amount Due (30)	***$1,440.00***
Amount Due (60)	$0.00
Amount Due (90)	***$196.00***

Customer Orders

Order # 1 ▾	Order Date	Total Amount
SO00003336	1/14/2011	$64.99
SO00003330	1/6/2011	$12.40
SO00003265	12/21/2010	$4,480.00
SO00003132	11/22/2010	$161.10
SO00003106	11/16/2010	$2,718.40
SO00003096	11/12/2010	$1,470.00
SO00003035	10/25/2010	$1,440.00
SO00002981	10/12/2010	$4,840.32
SO00002926	9/28/2010	$21.50
SO00002910	9/24/2010	$198.40
SO00002877	9/14/2010	$2,600.00
SO00002820	9/6/2010	$102.00

Figure 19: An example detail with a data annotation highlighting suspended status in red and amount due fields with bold and italics

Multiple Data Annotations are allowed for a Detail. Each Data Annotation can affect either one or more individual Properties or the entire section. When multiple Data Annotations affect the same Property or section, the last Annotation in the list will be applied. To change the order, select a Data Annotation and press the Move Up and Move Down toolbar buttons.



Detail Property Section

Data Object Detail	Details
Section Name	Property Section
Label	
Maintainable	Yes
Save Button Text	Save

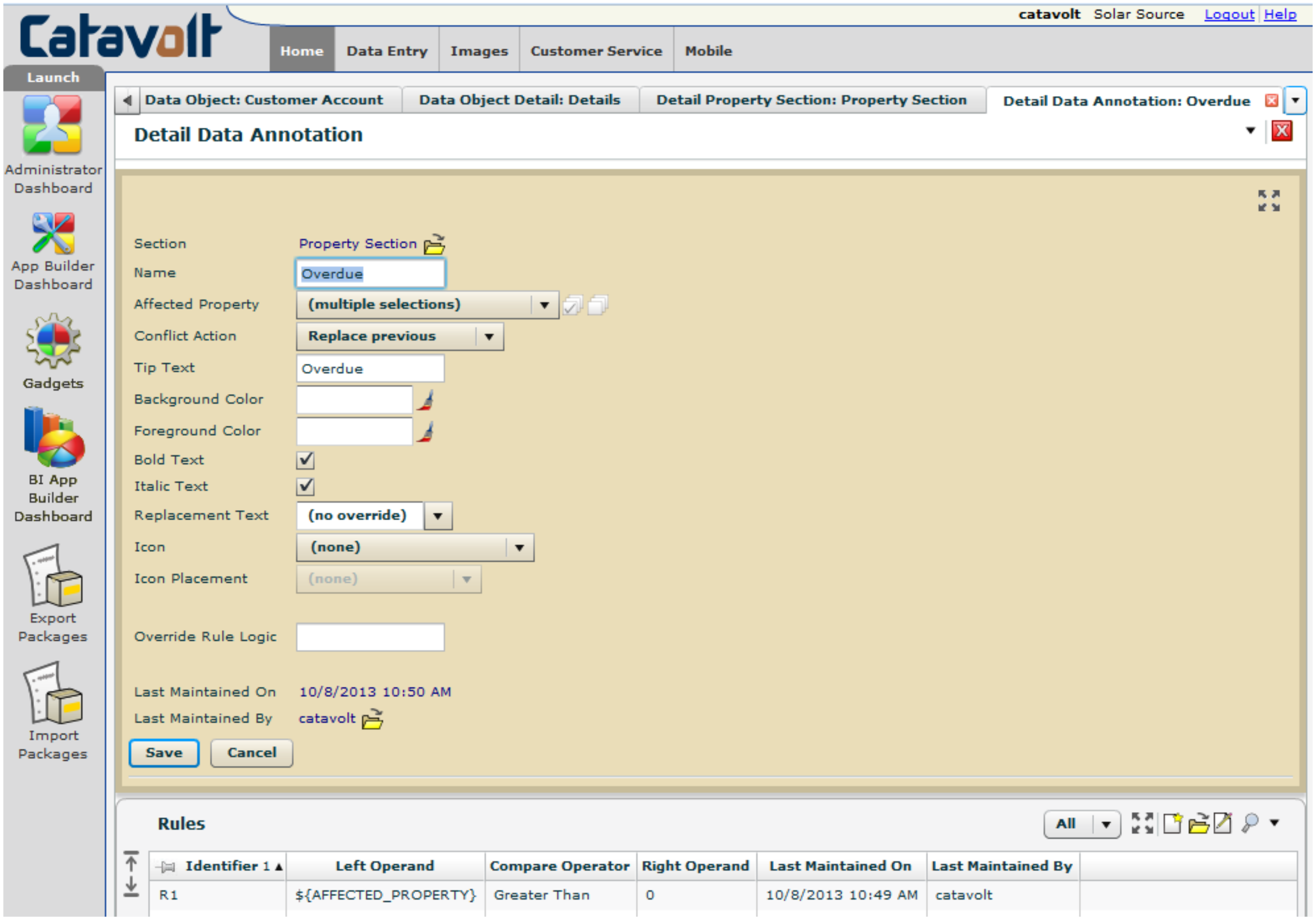
Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacement Text	Tip Text	Icon Placement
Suspended	**Sample**		#ff0000	Yes	No		Suspended	
Overdue	***Sample***			Yes	Yes		Overdue	

Properties

Name	Data Type	Conversion	Label	Maintainable	Style
account_number	String		Account #	No	Data
account_name	String		Name	No	Data
status	String		Status	Yes	Data
address1	String		Address	No	Data
city	String		City	Yes	Data
state	String		State	Yes	Data
postal_code	String		Zip	Yes	Data
amount_due_30	Decimal		Amount Due (30)	No	Data
amount_due_60	Decimal		Amount Due (60)	No	Data
amount_due_90	Decimal		Amount Due (90)	No	Data

Figure 20: The Detail Property Section details view highlighting the Data Annotations section



Section	Property Section
Name	Overdue
Affected Property	(multiple selections)
Conflict Action	Replace previous
Tip Text	Overdue
Background Color	
Foreground Color	
Bold Text	☑
Italic Text	☑
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)
Override Rule Logic	
Last Maintained On	10/8/2013 10:50 AM
Last Maintained By	catavolt

Identifier 1	Left Operand	Compare Operator	Right Operand	Last Maintained On	Last Maintained By
R1	${AFFECTED_PROPERTY}	Greater Than	0	10/8/2013 10:49 AM	catavolt

Figure 21: The create Detail Data Annotations view

When creating a Data Annotation, you must specify a Name. This value is used to uniquely identify the Data Annotation and is not presented to the end user.

Affected Property specifies which Property(s) will be altered by the Data Annotation or if it applies to the whole section. Xalt defaults this value to (Entire Row), which will alter every Property in the section. You also have the option of choosing one or more Properties from the Detail Property Section.

Conflict Action specifies what should be done if multiple Data Annotations apply to a property. As stated above, Annotations are processed in a specific sequence. Two annotations are in conflict of each other when they both pass all their rules and are annotating the same property or the entire section. When annotations are in conflict, the last one in the list wins. This option allows the annotation to either be merged with previous annotations or replace previous annotations. When an annotation is merged with a previous annotation, it will assume all previous annotations' properties that it does not specify itself. For example, if one annotation sets the background color and a latter one specifies a foreground color and they are merged, the resulting annotation will have both a background and a foreground color set.

Tip Text specifies text that will be displayed when the end user hovers over a property currently affected by a Data Annotation. In this way, the end user can get a text description of the Data Annotation that is currently being applied to the row or property without having to remember what the different colors, fonts, etc. mean when looking at a Query.

Background Color specifies the color you wish to change the background of the affected cell or the entire section if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Foreground Color specifies the color you wish to change the foreground text of the affected cell or the entire section if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Bold Text specifies whether you want to change the font to be bold. Select the checkbox to apply a bold font to the affected cell or the entire section if the Data Annotation is applied.

Italic Text specifies whether you want to change the font to be italicized. Select the checkbox to apply an italicized font to the affected cell or the entire section if the Data Annotation is applied.

Replacement Text specifies whether you want to replace the value of the affected cell with an alternate text string. This is commonly used to improve readability of the Detail (for example, to replace 0 values with blanks). Note that Replacement Text is not available when the Affected Property is (Entire Row).

Icon specifies whether you want to append or replace the value of the affected cell with an image. The image you choose can be shown instead of or in addition to the value of the affected cell. Note that Icon is not available when the Affected Property is (Entire Row).

Icon Placement specifies where to place the image in relation to the value of the affected cell. The available values are:

- Replace Text – Replace the value of the affected cell with the image.
- Left of Text – Put the image to the left of the value of the affected cell.
- Right of Text – Put the image to the right of the value of the affected cell.
- Background – Put the image in the background under the value of the affected cell. This value is not used on mobile clients.
- Background (Fill) – Put the image in the background under the value of the affected cell and stretch the image to fill the column horizontally. This is typically used for annotations that show progress bars or something similar. This value is not used on mobile clients

Override Rule Logic specifies more advanced logic to evaluate the Data Annotation Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Data Annotation to be applied. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

(R1 AND R2) OR (R3 AND R4)

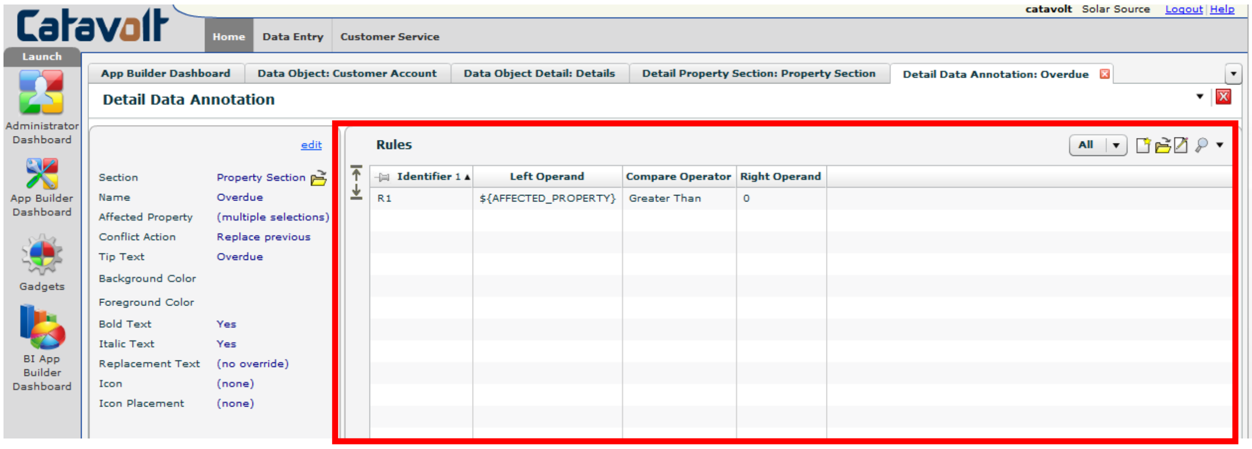
NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

Data Annotation Rules

The Rules list on a Data Annotation can be used to control when the Data Annotation should be applied to properties. A Data Annotation can have zero or more rules. All rules must pass for the Data Annotation to be applied to that particular row of data. If a Data Annotation has no rules, then it is considered to pass automatically.


Catavolt
catavolt Solar Source Logout | Help
Home
Data Entry
Customer Service
Launch
Administrator Dashboard
App Builder Dashboard
Gadgets
BI App Builder Dashboard
App Builder Dashboard
Data Object: Customer Account
Data Object Detail: Details
Detail Property Section: Property Section
Detail Data Annotation: Overdue
edit
All

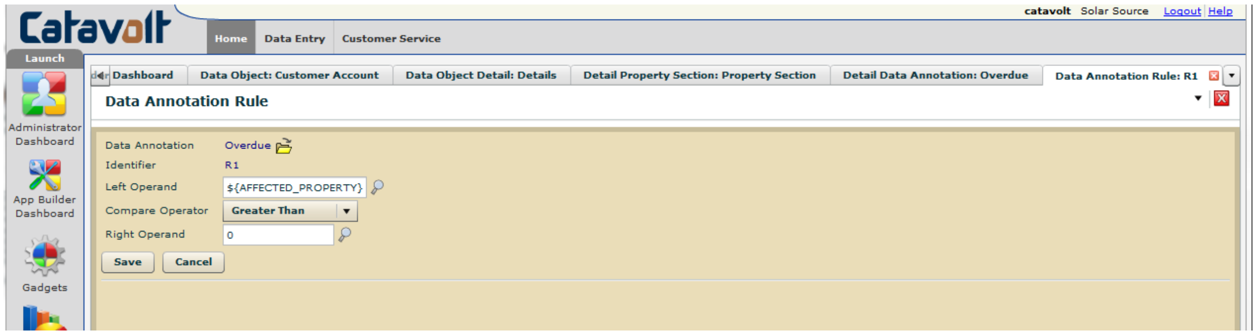
Detail Data Annotation

Section	Property Section
Name	Overdue
Affected Property	(multiple selections)
Conflict Action	Replace previous
Tip Text	Overdue
Background Color	
Foreground Color	
Bold Text	Yes
Italic Text	Yes
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand
R1	${AFFECTED_PROPERTY}	Greater Than	0

Figure 22: Detail Data Annotation details view with the Rules query section highlighted


Catavolt
catavolt Solar Source Logout | Help
Home
Data Entry
Customer Service
Launch
Administrator Dashboard
App Builder Dashboard
Gadgets
Dashboard
Data Object: Customer Account
Data Object Detail: Details
Detail Property Section: Property Section
Detail Data Annotation: Overdue
Data Annotation Rule: R1
Save
Cancel

Data Annotation Rule

Data Annotation	Overdue
Identifier	R1
Left Operand	${AFFECTED_PROPERTY}
Compare Operator	Greater Than
Right Operand	0

Figure 23: The create Data Annotation Rule view

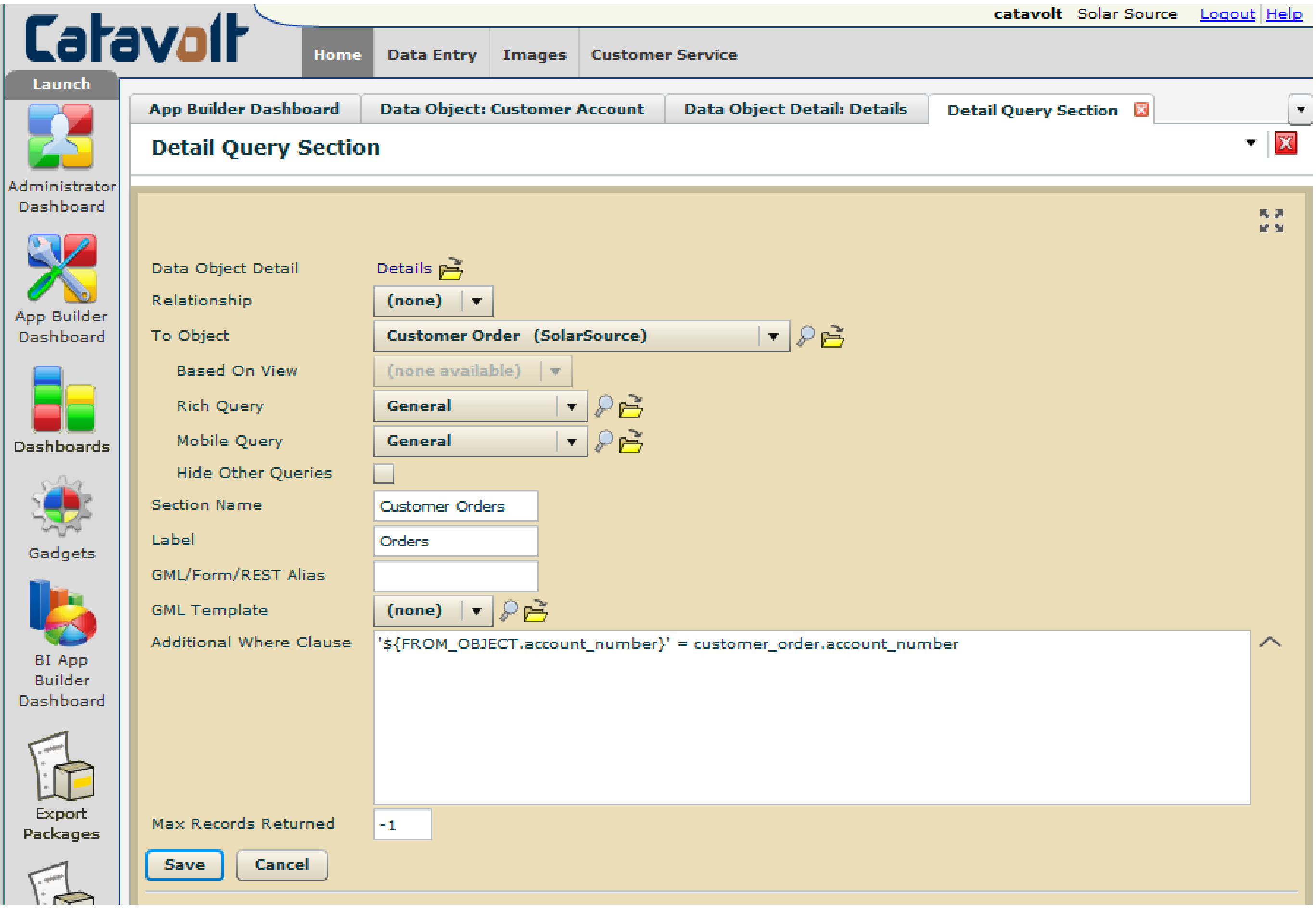
When creating a Data Annotation Rule, the Xalt will automatically create a unique Identifier for the Rule.

Left Operand and Right Operand specify the two values that will be compared with each other. You have 2 choices for operands: Constant Values (such as A or 1.00) or Substitution Values. You can enter a Substitution Value manually or press the Find button to have Xalt build a Substitution Value based on the Available Properties for the Data Object. Appendix A: Specifying Messages and Substitution Values for more information about how to create Substitution Messages for Left Operand and Right Operand. A special Substitution Value of ${AFFECTED_PROPERTY} is available for annotations where you select multiple Affected Properties. When running the annotation, the rule will be checked independently for each affected property. This allows you to create a single Annotation and a single rule that can apply to multiple individual properties. For example, if you have 5 numeric properties in a Detail Property Section and you want any negative values to show as red, you can create a single annotation to make the text foreground color red, choose all 5 numeric columns as Affected Properties, and create a single Annotation Rule of ${AFFECTED_PROPERTY} Less Than 0.

Compare Operator specifies how the two operands should be compared with each other. Options include Equal, Not Equal, Greater Than, Less Than, Greater or Equal, Less or Equal, Contains, Starts With, Ends With, Is Null, and Is Not Null.

Query Section

A Query Section will display a list of records in a section for the given Data Object. This list is typically (but not always) related to the Data Object in which it is being displayed.



Detail Query Section

Data Object Detail	Details
Relationship	(none)
To Object	Customer Order (SolarSource)
Based On View	(none available)
Rich Query	General
Mobile Query	General
Hide Other Queries	□
Section Name	Customer Orders
Label	Orders
GML/Form/REST Alias	
GML Template	(none)
Additional Where Clause	'${FROM_OBJECT.account_number}' = customer_order.account_number
Max Records Returned	-1

Figure 24: The create Detail Query Section view

Relationship specifies the relationship from the given Data Object that is traversed to display the list of related records. Some back-end systems (such as ODATA and IDF System-Link) have predefined relationships that it can traverse to display related records (e.g. All Purchase Order Items for a specified Purchase Order) with no extra effort on your part. If the Data Object Allows Attachments, an (attachment) option will be available to show a list of Attachments for the record (See Appendix D: Attachments for more information on using Attachments). If no predefined relationships exist or your back-end system does not support this feature, (none) will be displayed. Note that even if predefined relationships exist, you can still select (none) to create an ad-hoc related list.

To Object specifies which Data Object you will display a list for. If you selected a Relationship, this list will contain all defined Data Objects whose class matches that of the Relationship, plus a (create new) option that will create a new Data Object for the specified class of the Relationship. If you selected (attachment) for the Relationship, this list will contain all Attachments objects defined for you Data Source. If you selected (none) for the Relationship, this list will contain all Data Objects defined for your Data Source.

Based On View specifies which predefined view you would like to use when creating a Query. Note that this option is only available when you select (create new) as your To Object and if your back end data source supports predefined views.

Rich Query specifies which Query you want to use when displaying this section on a Rich client. Note that this option is only available when you select a To Object other than (create new).

Mobile Query specifies which Query you want to use when displaying this section on a Mobile client. Note that this option is only available when you select a To Object other than (create new).

Hide Other Queries specifies whether the runtime Query dropdown list will contain just the selected Rich/Mobile Query or all non-hidden Queries. If Hide Other Queries is selected, only the selected Rich/Mobile Query will be available to choose in the Query dropdown list.

Section Name is used to uniquely identify the Property Section and is not presented to the end user. Xalt defaults this value to the Object Name (Plural) of the selected To Object.

Label specifies an identifying description of the section as displayed to the end user in the top left corner. Leaving this value empty will cause no label to be displayed for the section. Xalt defaults this value to the Object Name (Plural) of the selected To Object.

GML/Form Alias allows you to specify an alias when using this Section with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Additional Where Clause specifies additional criteria that can be used to further filter the list of records to be displayed. As described above, if you specified a Relationship, your list will already be filtered based on the record being displayed. In this case, you could specify an additional where to supplement the where that is implied by the relationship, for example showing only open order line items. If you specified (none) for Relationship and you do not specify an Additional Where Clause, you will get an unfiltered list, for example all order line items for all orders. In most cases, you need to specify an Additional Where Clause to manually restrict the list to just the Order Items for the specified Order (plus any extra criteria that you may require). See Appendix B: Specifying Where Clauses for more information about specifying Additional Where Clause values.

Max Records Returned specifies the number of records to return in the list. Xalt defaults this value to -1, which will return all records.

Please note that query sections for IDF System-Link data sources that use relationships do not support paging of data and thus will read all records at once. To prevent adverse performance effects of this the number of records read will be limited to 1000 when a Max Records Returned of -1 is used.

Visualization Section

A Visualization Section will display a File or Web-API based visualization of the selected ODATA detail record:

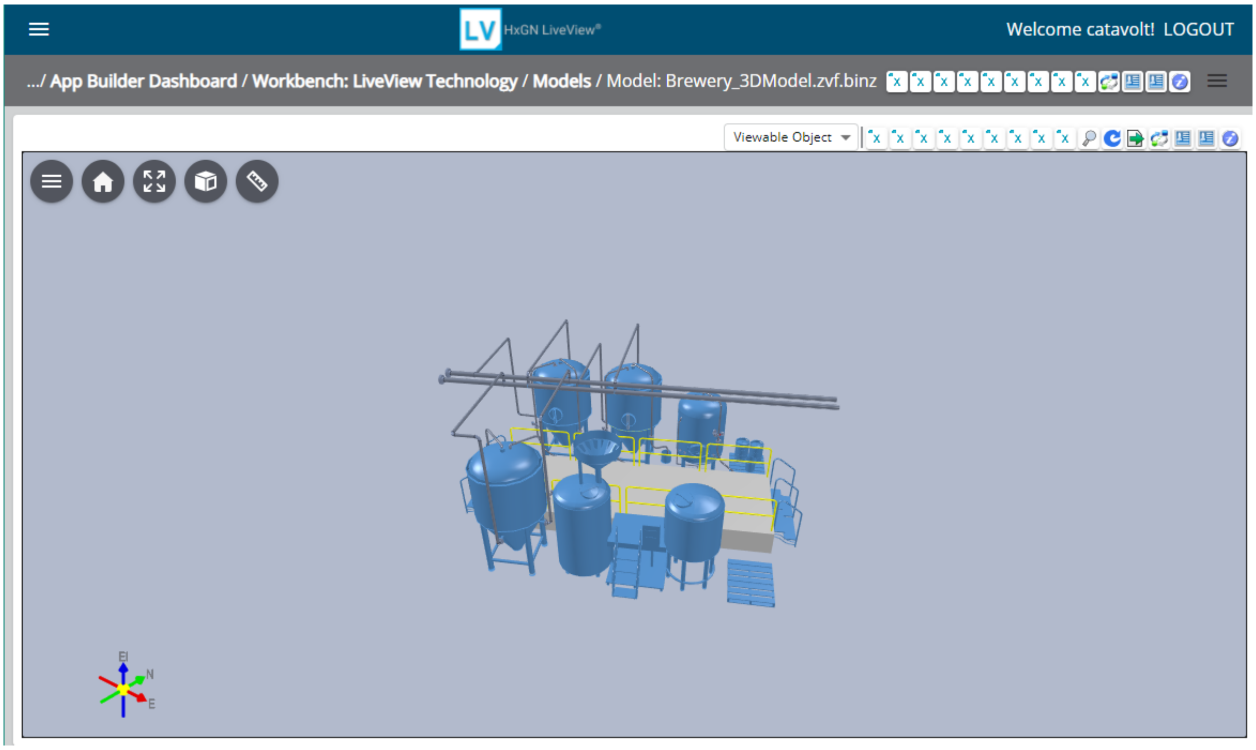
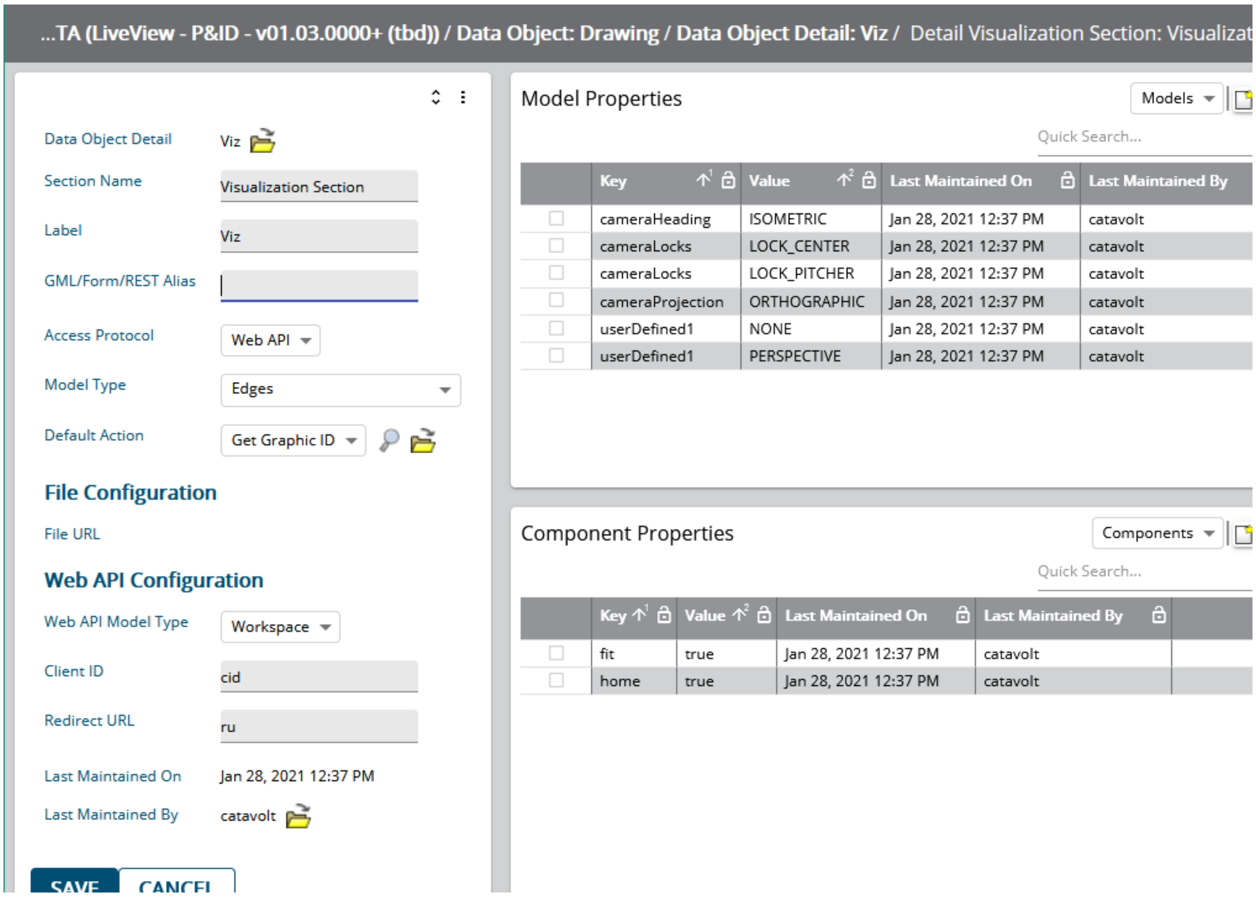

LV HxGN LiveView®
Welcome catavolt! LOGOUT
.../ App Builder Dashboard / Workbench: LiveView Technology / Models / Model: Brewery_3DModel.zvf.binz
Viewable Object
El
N
E

Figure 25: The Detail Visualization Section runtime view



...TA (LiveView - P&ID - v01.03.0000+ (tbd)) / Data Object: Drawing / Data Object Detail: Viz / Detail Visualization Section: Visualizat

Data Object Detail	Viz
Section Name	Visualization Section
Label	Viz
GML/Form/REST Alias	
Access Protocol	Web API
Model Type	Edges
Default Action	Get Graphic ID

File Configuration

File URL	

Web API Configuration

Web API Model Type	Workspace
Client ID	cid
Redirect URL	ru
Last Maintained On	Jan 28, 2021 12:37 PM
Last Maintained By	catavolt

Model Properties

	Key	Value	Last Maintained On	Last Maintained By
□	cameraHeading	ISOMETRIC	Jan 28, 2021 12:37 PM	catavolt
□	cameraLocks	LOCK_CENTER	Jan 28, 2021 12:37 PM	catavolt
□	cameraLocks	LOCK_PITCHER	Jan 28, 2021 12:37 PM	catavolt
□	cameraProjection	ORTHOGRAPHIC	Jan 28, 2021 12:37 PM	catavolt
□	userDefined1	NONE	Jan 28, 2021 12:37 PM	catavolt
□	userDefined1	PERSPECTIVE	Jan 28, 2021 12:37 PM	catavolt

Component Properties

	Key	Value	Last Maintained On	Last Maintained By
□	fit	true	Jan 28, 2021 12:37 PM	catavolt
□	home	true	Jan 28, 2021 12:37 PM	catavolt

Figure 26: The Detail Visualization Section view

Section Name is used to uniquely identify the Property Section and is not presented to the end user. Xalt defaults this value to Visualization Section.

Label specifies an identifying description of the section as displayed to the end user in the top left corner. Leaving this value empty will cause no label to be displayed for the section. Xalt defaults this value to the Object Name (Plural) of the selected To Object.

GML/Form Alias allows you to specify an alias when using this Section with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Access Protocol specifies the protocol used to retrieve Visualization information. The available values are File and Web API. Note that (Web API is only available if the Data Object in question inherits from or implements Com.Ingr.Core.V1.ViewableObject).

Model Type specifies the type of model to be displayed.

Default Action specifies the Action to be performed when the user selects an item inside the Visualization Section.

File URL specifies the URL location of the Visualization file (used when Access Protocol is set to File). This value be hardcoded or can use Substitution Values.

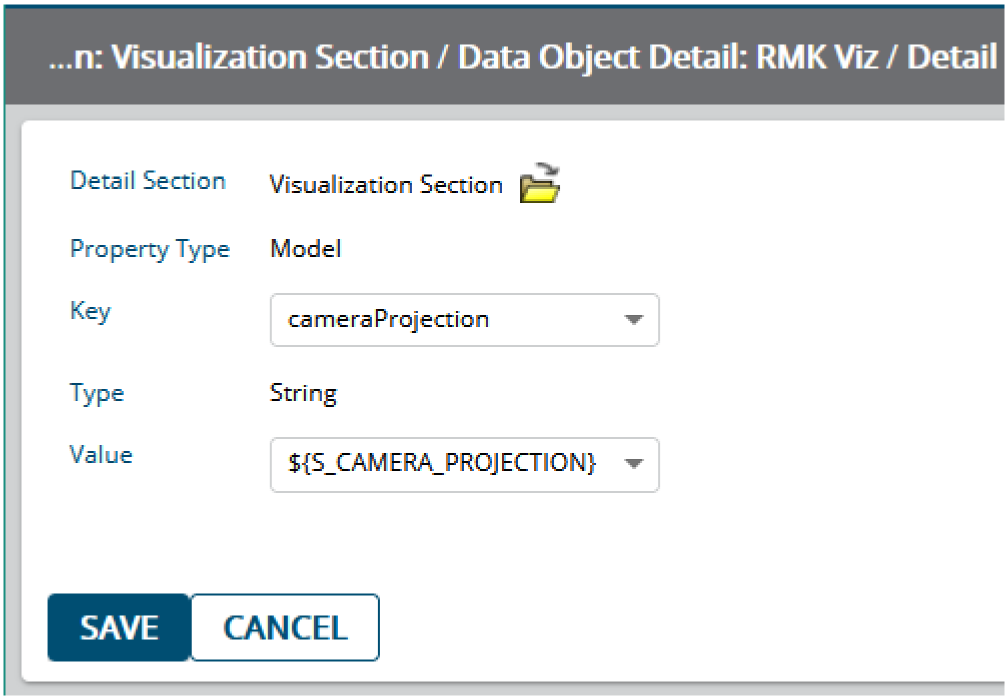
Web API Model Type specifies the type of model to be used in Visualization (used when Access Protocol is set to Web API). The available values are File and Workspace.

Client ID specifies the OAUTH PKCE Client ID used to authenticate with the Visualization GDS server.

Redirect URL specifies the OAUTH Redirection URL used during authentication with the Visualization GDS server. The default value of https://xha.hexagonxalt.net/gvcAuthRedirect.html should not need to be changed in normal circumstances.

Model / Component Properties

The Visualization Section accepts two sets of configuration properties, Model and Component



...n: Visualization Section / Data Object Detail: RMK Viz / Detail

Detail Section: Visualization Section

Property Type: Model

Key: cameraProjection

Type: String

Value: ${S_CAMERA_PROJECTION}

Figure 27: The Detail Visualization Section Property view

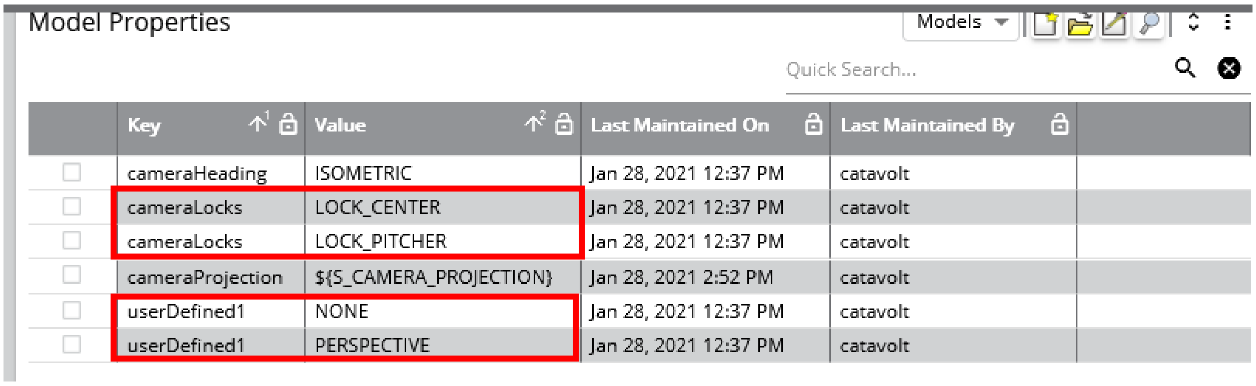
Property Type specifies whether this is a Model or a Component Property.

Key is used to identify the Property Key. Extender provides a Combo Box containing a list of common keys to choose from, or you may enter your own value to be passed to the Visualization Section.

Type specifies the Data type of the Property. Available values are Boolean, Number, and String. If you select a common value for Key, the appropriate value will automatically be selected.

Value specifies the Property Value. You may enter a constant value or use a Substitution Property for the value. If you select a common value for Key, a list of common Values for that key will be available to select from.

Note that the **cameraLocks** and **defaultSettings** Keys (as well as user-defined Keys) accept multiple values. For these Keys, you may create multiple Properties with the same Key and different values. At runtime, Extender will combine the values together and pass them to the Visualization Section as a single multi-value Property.



Model Properties

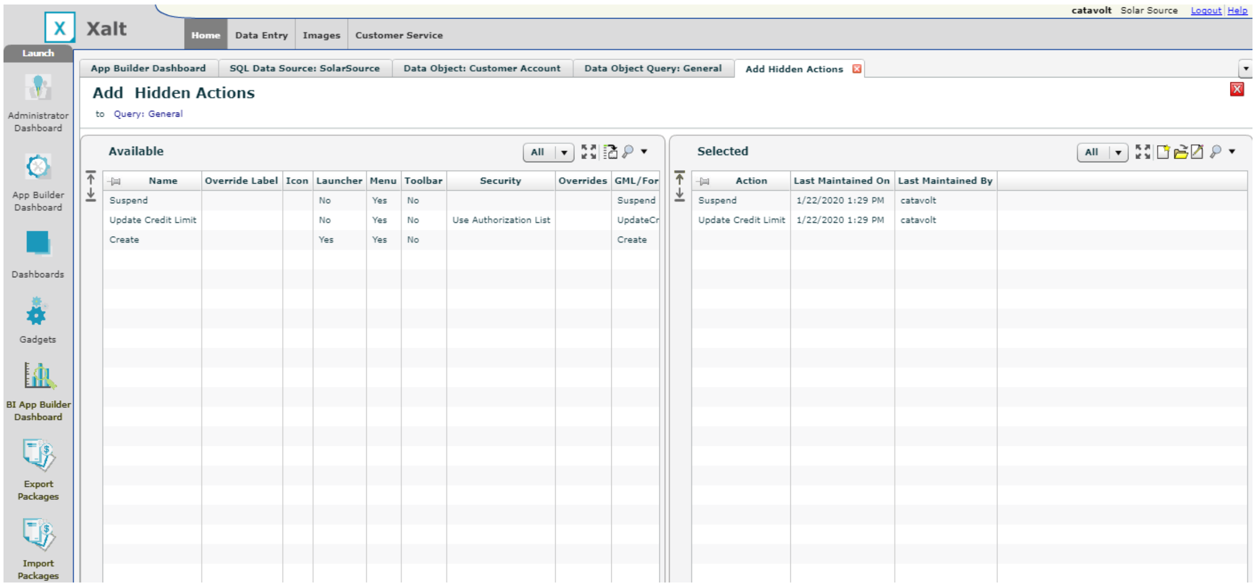
Quick Search...

	Key	Value	Last Maintained On	Last Maintained By
□	cameraHeading	ISOMETRIC	Jan 28, 2021 12:37 PM	catavolt
□	cameraLocks	LOCK_CENTER	Jan 28, 2021 12:37 PM	catavolt
□	cameraLocks	LOCK_PITCHER	Jan 28, 2021 12:37 PM	catavolt
□	cameraProjection	${S_CAMERA_PROJECTION}	Jan 28, 2021 2:52 PM	catavolt
□	userDefined1	NONE	Jan 28, 2021 12:37 PM	catavolt
□	userDefined1	PERSPECTIVE	Jan 28, 2021 12:37 PM	catavolt

Figure 28: The Detail Visualization Section Property view showing multiple Properties with the same Key

Hidden Actions

Hidden Actions specifies a list of Actions that should not be shown on the menu when this Detail is active (if you have set the Menu Actions property to Use Hidden Actions List). When adding Hidden Actions, you will be presented with two lists. The Available Actions list shows all actions for the Data Object (including both custom actions as well as override standard actions). The Selected Actions list shows the actions that will be hidden on the menu.



Add Hidden Actions

to Query: General

Available

Name	Override Label	Icon	Launcher	Menu	Toolbar	Security	Overrides	GML/For
Suspend			No	Yes	No			Suspend
Update Credit Limit			No	Yes	No	Use Authorization List		UpdateCr
Create			Yes	Yes	No			Create

Selected

Action	Last Maintained On	Last Maintained By
Suspend	1/22/2020 1:29 PM	catavolt
Update Credit Limit	1/22/2020 1:29 PM	catavolt

Figure 29: The Add Hidden Actions view

Excluding an Action from the menu will mark the action as Hidden. It can still be called directly from GML or Custom Forms, same as with Hidden actions.

The available list of Hidden Actions includes all custom actions in the Action list for the Data Object. If the you want to exclude a standard action (such as Export, Refresh, etc.), you should first run Override Standard Action and add it to the Actions list. At that point the Action will be available to add to the Hidden Actions list.

Since a menu is tied to a specific Query/Detail, if you have a Detail Query Section / Dashboard Query Section / Open List Completion Action and you want to have a special menu just for that section, you should create a dedicated (hidden) Query just to be used for that section that contains a custom menu.

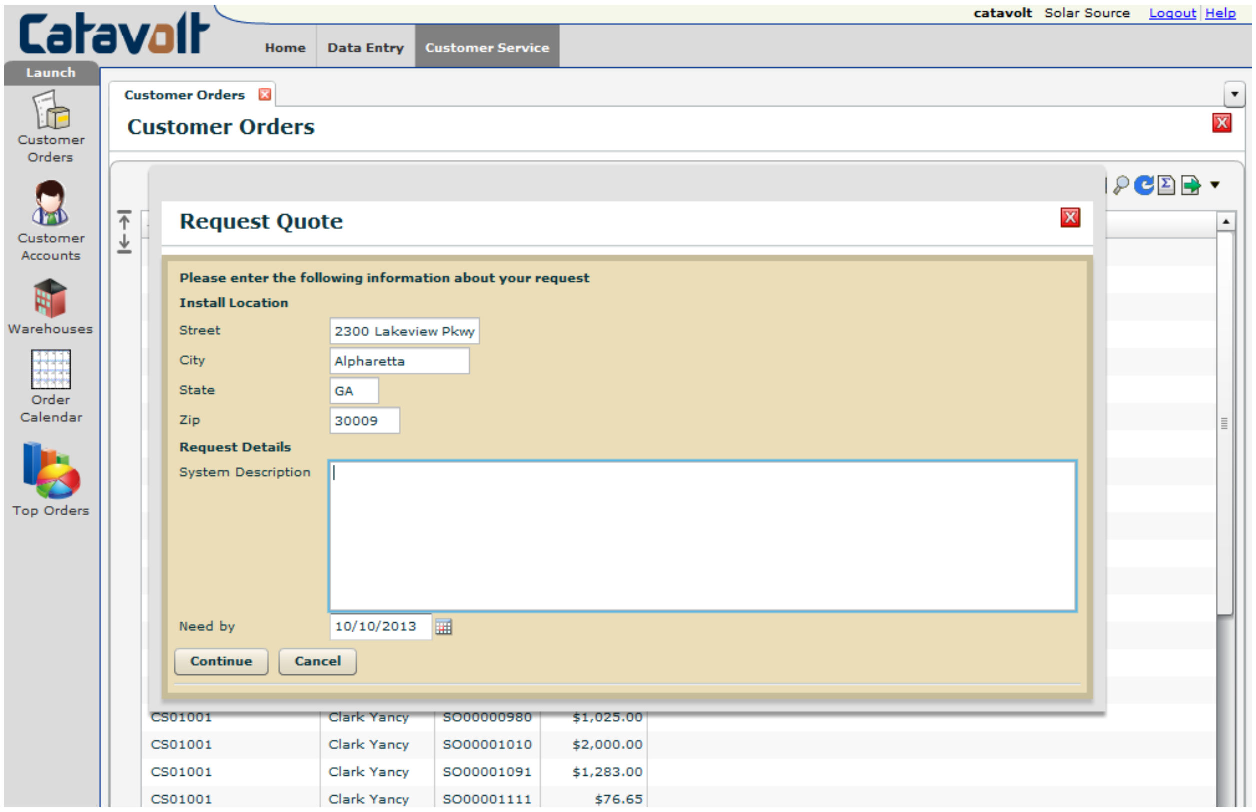
Chapter 6: Data Object Actions

Chapter Summary

A Data Object Action is an option that appears on the toolbar and context menu of a Data Object. When the option is executed it can prompt the user for input or confirmation and then execute an XML script of commands against the back end system. The available types of XML that can be executed varies depending on the type of Data Source.

The prompt for the Data Object Action can request input from the user. The prompt can be configured very similarly to a Property Section on a Data Object Detail. The prompt can contain constants, blank lines, and data properties. The data properties can have their values defaulted based on the selected Data Object record, system substitution values, data source substitution value, and/or user properties.

Optionally, after the XML has been performed for an Action, a Completion Action may be performed. This Completion Action can download a data file (PDF, image, etc) and display it to the end user. It can also open a new tab and display a list of Xalt data or the details of an Xalt object. For example, an Action on a Customer Order Line Item may end up creating a Manufacturing Order. Using the Open Object Completion Action, the created Manufacturing Order can be displayed after the Action has been performed.



Customer Orders

Request Quote

Please enter the following information about your request

Install Location

Street: 2300 Lakeview Pkwy

City: Alpharetta

State: GA

Zip: 30009

Request Details

System Description:

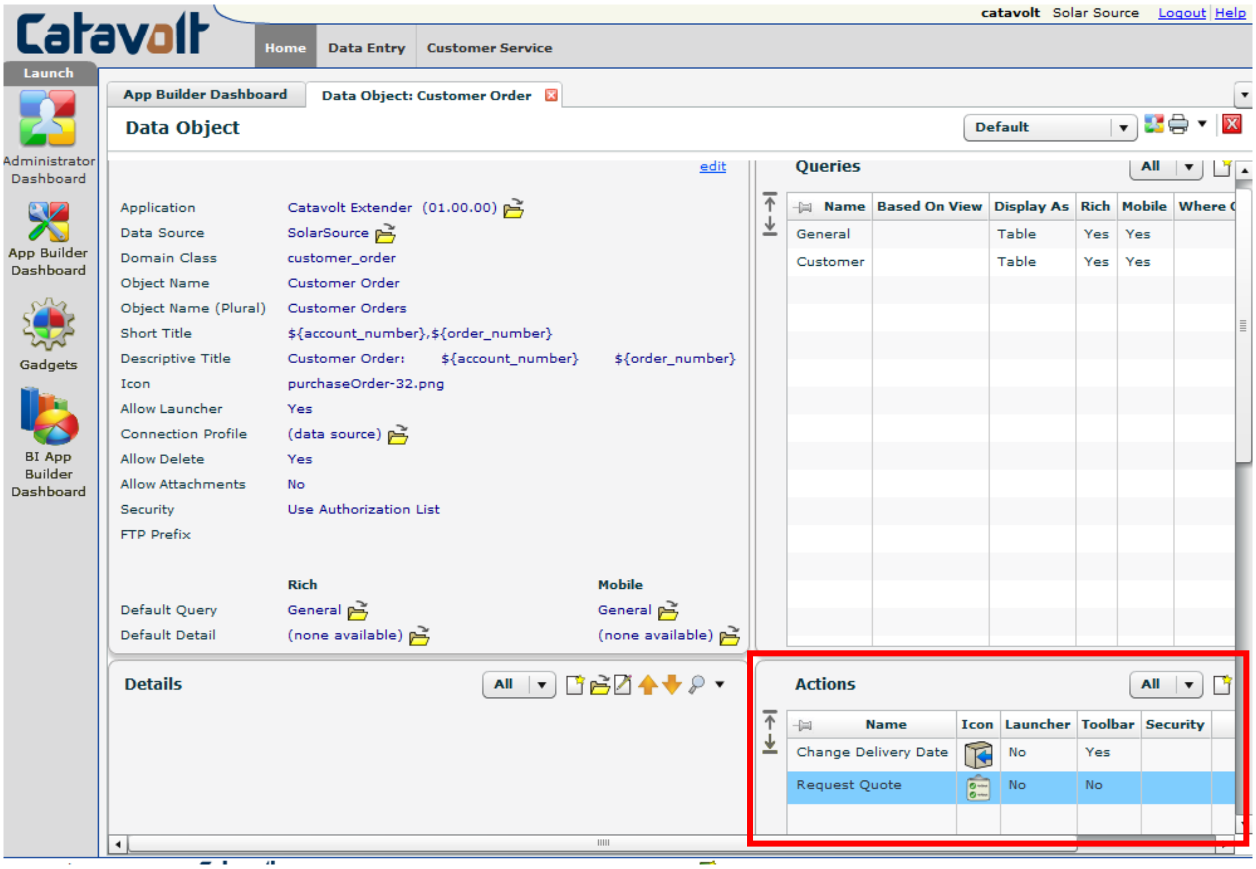
Need by: 10/10/2013

CS01001	Clark Yancy	SO00000980	$1,025.00
CS01001	Clark Yancy	SO00001010	$2,000.00
CS01001	Clark Yancy	SO00001091	$1,283.00
CS01001	Clark Yancy	SO00001111	$76.65

Figure 1: An example Data Object Action running in a web browser

Accessing Actions

To access the Actions defined for a particular Data Object, open the Data Object and look at the Default Detail. Multiple Actions can be created for a Data Object. They will appear on the toolbar and context menu in the same order they appear below. To change the order, select an Action and press the Move Up and Move Down toolbar buttons.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_order
Object Name	Customer Order
Object Name (Plural)	Customer Orders
Short Title	${account_number},${order_number}
Descriptive Title	Customer Order: ${account_number} ${order_number}
Icon	purchaseOrder-32.png
Allow Launcher	Yes
Connection Profile	(data source)
Allow Delete	Yes
Allow Attachments	No
Security	Use Authorization List
FTP Prefix	

	Rich	Mobile
Default Query	General	General
Default Detail	(none available)	(none available)

Queries

Name	Based On View	Display As	Rich	Mobile	Where (
General		Table	Yes	Yes	
Customer		Table	Yes	Yes	

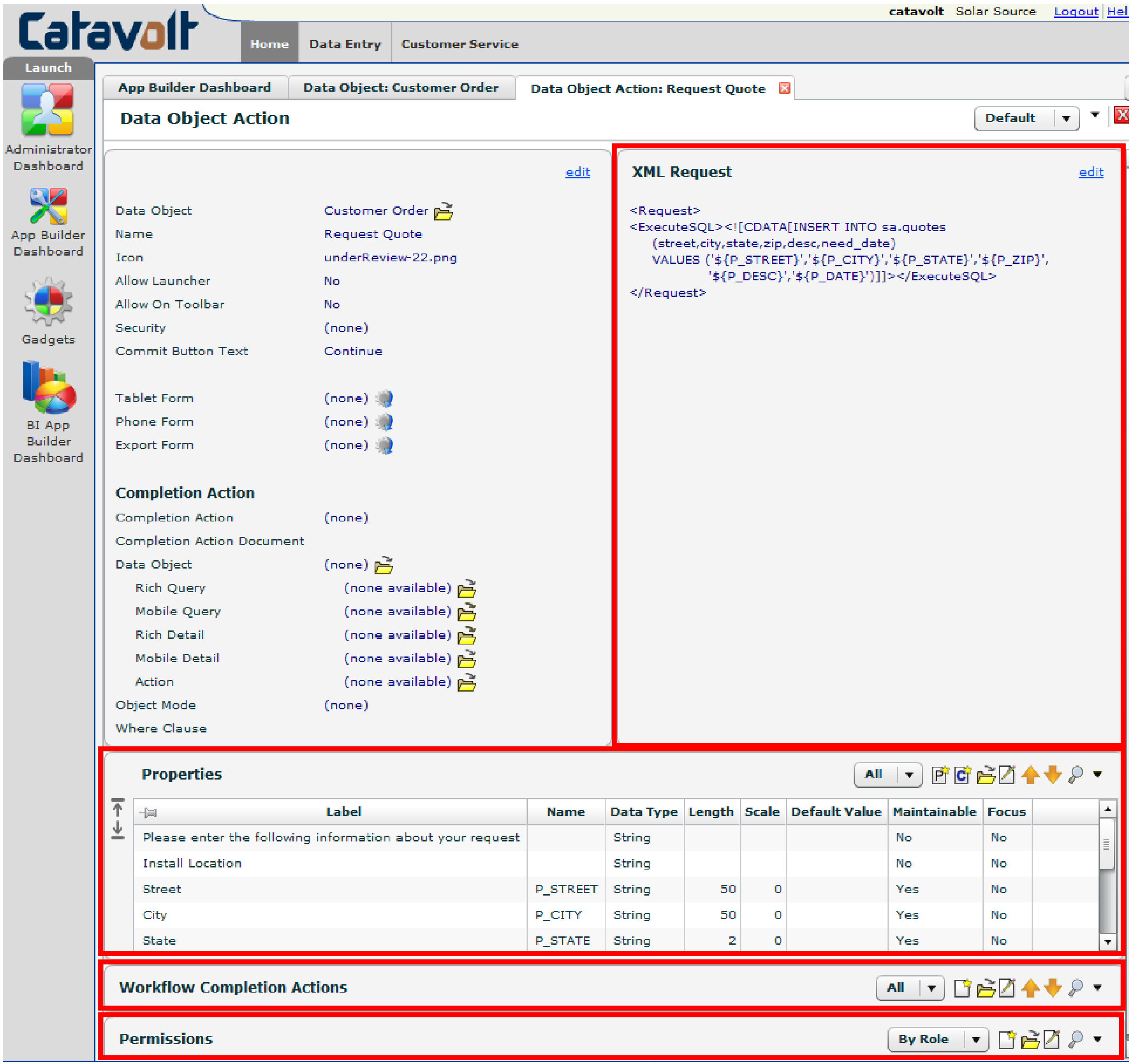
Actions

Name	Icon	Launcher	Toolbar	Security
Change Delivery Date		No	Yes	
Request Quote		No	No	

Figure 2: The Data Object Default details view with the Actions query section highlighted

Action Components

When you display the definition for an Action, there are 2 Details that you can select from. The Default Detail shows you all the information about the Action. An Action is made up of 4 components: XML Request (the work that is to be performed on the back end system), Properties (a list of prompted values provided by the end user), Permissions (security on the action) and Workflow Completion Actions (a Workflow to determine the Completion Action after the action is performed).



Data Object Action

Data Object	Customer Order
Name	Request Quote
Icon	underReview-22.png
Allow Launcher	No
Allow On Toolbar	No
Security	(none)
Commit Button Text	Continue
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)

Completion Action

Completion Action	(none)
Completion Action Document	
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Action	(none available)
Object Mode	(none)
Where Clause	

XML Request

```
<Request>
<ExecuteSQL><![CDATA[INSERT INTO sa.quotes
    (street,city,state,zip,desc,need_date)
    VALUES ('${P_STREET}','${P_CITY}','${P_STATE}','${P_ZIP}',
            '${P_DESC}','${P_DATE}')]]></ExecuteSQL>
</Request>
```

Properties

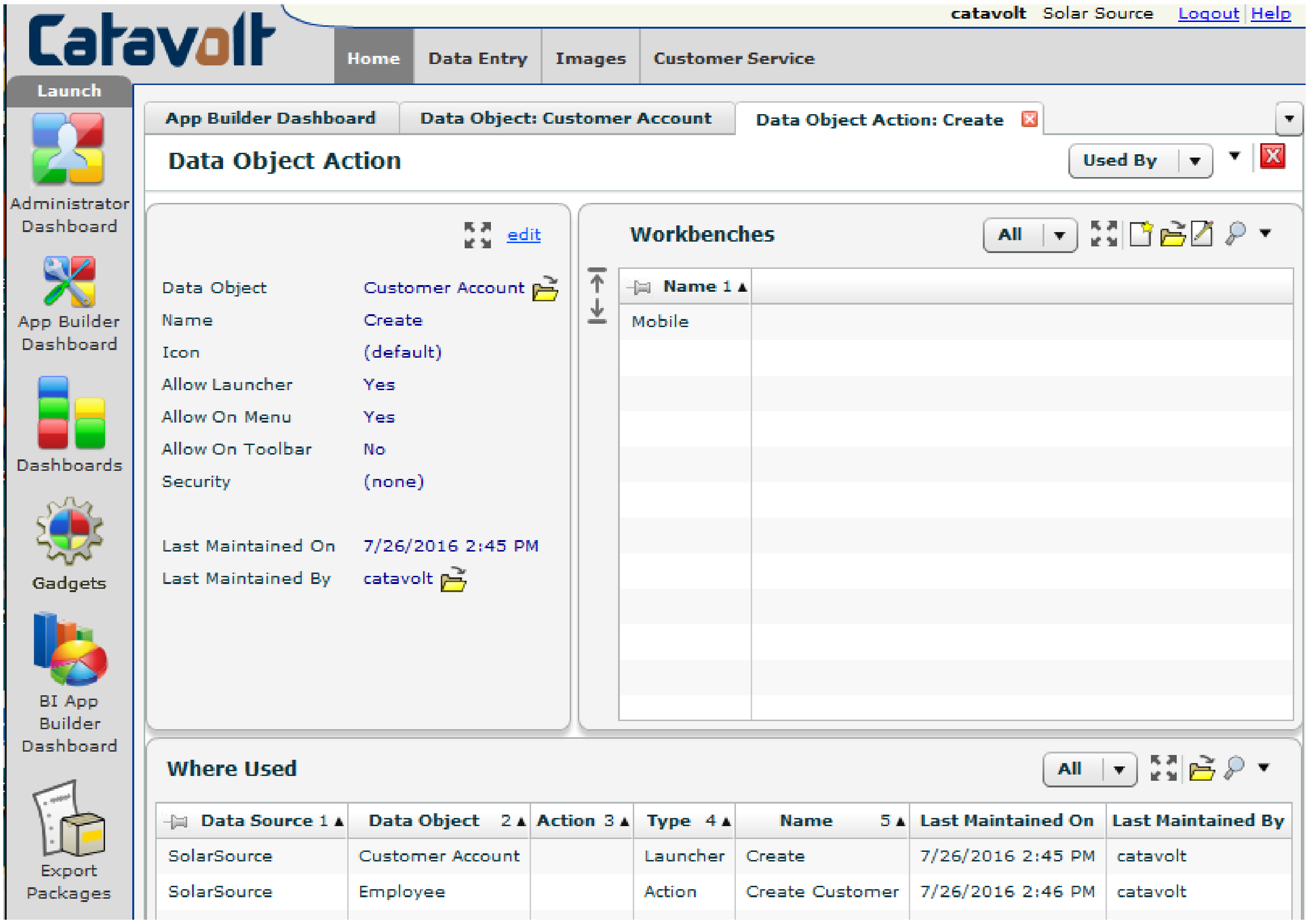
Label	Name	Data Type	Length	Scale	Default Value	Maintainable	Focus
Please enter the following information about your request		String				No	No
Install Location		String				No	No
Street	P_STREET	String	50	0		Yes	No
City	P_CITY	String	50	0		Yes	No
State	P_STATE	String	2	0		Yes	No

Workflow Completion Actions

Permissions

Figure 3: A Data Object Action details view with the XML Request, Properties, Workflow Completion Actions, and Permissions sections highlighted

The Used By Detail shows a list of all Workbenches that currently have this Action on them as well as a Where Used list of objects that are currently using this Action.



Data Object Action

Data Object	Customer Account
Name	Create
Icon	(default)
Allow Launcher	Yes
Allow On Menu	Yes
Allow On Toolbar	No
Security	(none)
Last Maintained On	7/26/2016 2:45 PM
Last Maintained By	catavolt

Workbenches

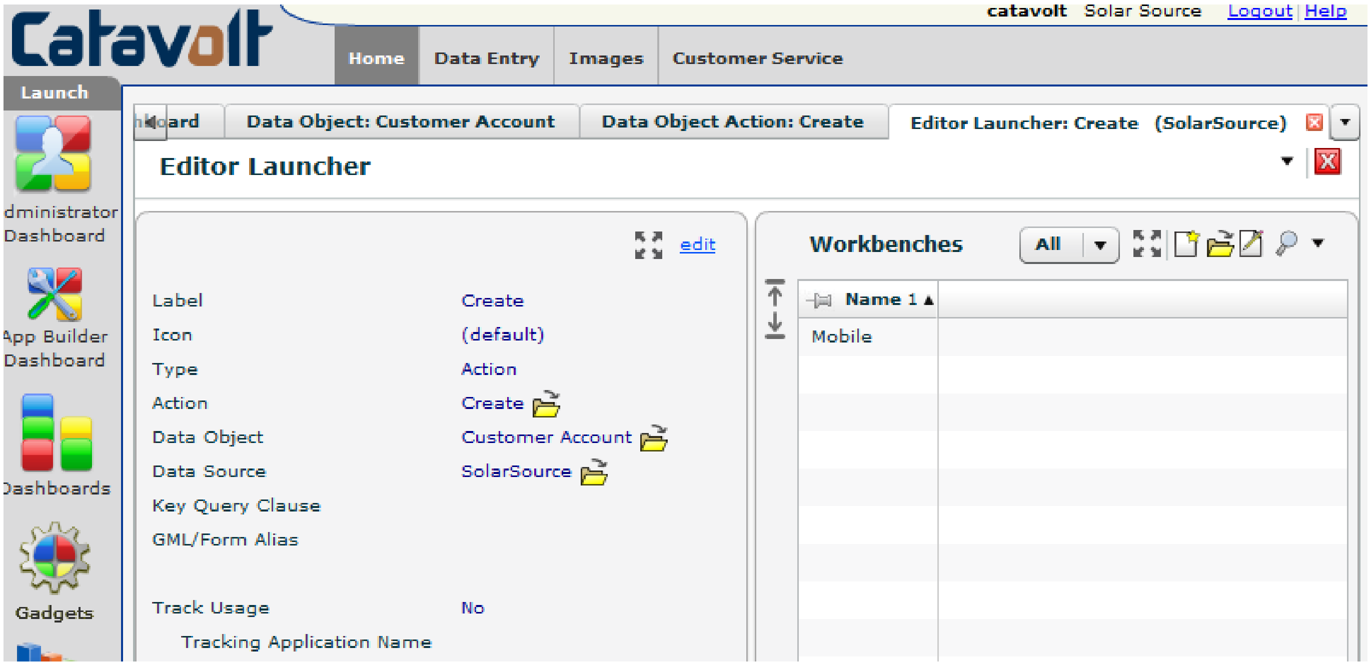
Name 1 ▲
Mobile

Where Used

Data Source 1 ▲	Data Object 2 ▲	Action 3 ▲	Type 4 ▲	Name 5 ▲	Last Maintained On	Last Maintained By
SolarSource	Customer Account		Launcher	Create	7/26/2016 2:45 PM	catavolt
SolarSource	Employee		Action	Create Customer	7/26/2016 2:46 PM	catavolt

Figure 4: Action 'Used By' details

Opening a Where Used record will take you directly to the object using the Action:



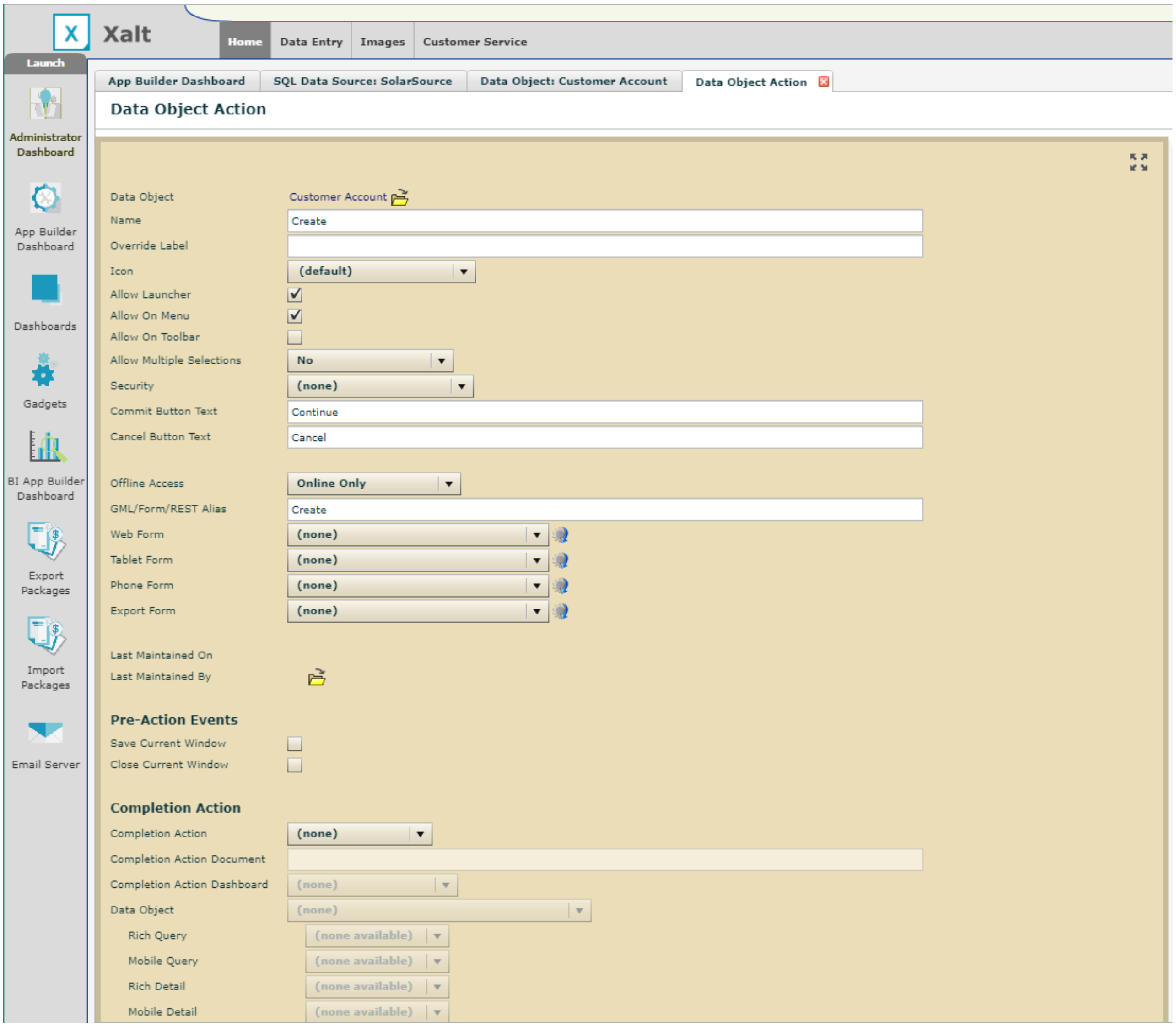
hboard | Data Object: Customer Account | Data Object Action: Create | Editor Launcher: Create (SolarSource)

Editor Launcher

Label	Create
Icon	(default)
Type	Action
Action	Create
Data Object	Customer Account
Data Source	SolarSource
Key Query Clause	
GML/Form Alias	
Track Usage	No
Tracking Application Name	

Figure 5: Action Where Used

Creating Actions



Data Object Action

Data Object	Customer Account
Name	Create
Override Label	
Icon	(default)
Allow Launcher	☑
Allow On Menu	☑
Allow On Toolbar	□
Allow Multiple Selections	No
Security	(none)
Commit Button Text	Continue
Cancel Button Text	Cancel
Offline Access	Online Only
GML/Form/REST Alias	Create
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	
Last Maintained By	

Pre-Action Events

Save Current Window	□
Close Current Window	□

Completion Action

Completion Action	(none)
Completion Action Document	
Completion Action Dashboard	(none)
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)

Figure 6: The create Data Object Action view

Name specifies the name of the Action.

Override Label specifies the value used as the text for the action on the context menu as well as the tip text for the action on the toolbar and the default Launcher name (if you specify Allow Launcher). If you leave this value blank, the Name value will be used instead. This is typically used in cases where you have multiple Actions that perform similar functions. You can use Name to differentiate the Actions for developers while using Override Label to make them more readable to the end user.

Icon specifies the image that will be used when displaying the action on the context menu and toolbar. You can choose the default Hexagon image or any other image you have uploaded (To upload an image, choose the Upload Image menu action from the Data Sources list).

Allow Launcher specifies whether to create a Launcher for this action. Creating a Launcher will allow you to run this Action directly from a Workbench without first having to navigate to a Data Object. Note that this option is only available if this Action does not contain any Data Object Substitution Values in the Completion Action Document, XML Request, or Properties. If any of these values contain Data Object Substitution Values, the user will be required to select a Data Object record when running this action, and you cannot select a Data Object record when running the action directly via a Launcher.

Allow On Menu specifies whether to add this action to the Data Object's menu. If you have actions that are being called directly from GML, Custom Forms, a Launcher, or as a result of a Completion Action, you may decide not to include it on the Data Object's menu.

Allow On Toolbar specifies whether to add this action to the Data Object's toolbar.

Allow Multiple Selections allows you to specify whether the user can select multiple records when performing this Action. The available options are:

- No – This Action supports single select only. Multiple selected records are not allowed.
- Yes (Multi-Prompt) – This Action supports multiple selected records. A prompt will be displayed for each selected record.
- Yes (Single Prompt) – This Action supports multiple selected records. A prompt will be displayed once only, and the Requested XML will be performed for each selected record.

See Allowing Multiple Selections below for a more comprehensive explanation of this function.

Security allows you to secure this Action to specific roles. The available options are:

- (none) – Do not add extra security this Action. Anyone who can launch or navigate to this Action will have access to perform it.
- Adopt Read Authority – Anyone who has read authority to this Data Object will have access to perform the Action.
- Adopt Update Authority – Anyone who has update authority to this Data Object will have access to perform the Action.
- Use authorization list – This option allows you to define a Permission list that contains one or more Security Roles that are allowed to perform the Action.

See Chapter 10: Security for more information about object security in Xalt.

Commit Button Text specifies the label on the Commit button when the Action prompt is displayed. The default value is Continue.

Cancel Button Text specifies the label on the Cancel button when the Action prompt is displayed. The default value is Cancel.

Offline Access allows you to specify whether this Action can be performed while offline. The available options are:

Perform Maintenance – This action will perform maintenance when performed offline

Retrieve Data – This action will retrieve data when performed offline

Online Only – Do not allow this action to be performed offline

Please see the **Xalt Mobility – Offline Guide** for more information about Offline Access.

GML/Form Alias allows you to specify an alias when using this Action with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Web Form specifies which custom form will be used to display the data for this detail when on a Web Browser.

Tablet Form specifies which custom form will be used to display the data for this detail when on a tablet device.

Phone Form specifies which custom form will be used to display the data for this detail when on a phone device.

Export Form specifies which custom form will be used to display the data for this detail when choosing the Export Form to PDF menu option. Selecting (none) will remove the Export Form to PDF menu option from this detail.

You can also view, upload, and remove custom forms from a dedicated launcher. See Chapter 13: Image/Asset Management for more information.

Save Current Window specifies whether the current window should be saved before running the action. If the save returns errors, the errors will be shown and the menu action will not be performed. If the save completes successfully, the menu action will then be performed. Note that the save will not be performed if the action is run on a Query View or against a Detail that is in read mode (as there are no pending changes to be saved). This option is typically used in cases where Actions have been specified to run as buttons on Custom Forms or in GML.

Close Current Window specifies whether the current window should be closed before running the action. This option is typically used in cases where Actions have been specified to run as buttons on Custom Forms or in GML.

Completion Action allows you to navigate to some other location when the Action has completed. Currently, the following values are allowed:

- Open Dashboard – Open Dashboard will open a new tab and display a Dashboard.
- Open Document – Open Document will download a specified document via FTP and display it in the client (if Allow FTP is set to Yes for the Action's Data Source). You may also specify a private URL that will be accessed via the Connector gateway and then downloaded and displayed by the client
- Open URL – Open URL will open a new browser tab and display the specified URL.
- Open List – Open List will open a new tab and display a Data Object Query.
- Open Object – Open Object will open a new tab and display a Data Object Detail in either read or update mode
- Open Action – Open Action will run the action you specify after the completion of the current action.
- Open EDE View – Open EDE View will open a Hexagon PPM EDE (Engineering Data Editor) View for display. This option is only available for Actions with an ODATA Data Source. Note: This option requires a <SetEDEConfiguration> section in your Request XML.
- Display Visualization Object Properties -- Display Visualization Object Properties allows you to display a list of properties for a selected object inside a Visualization Section. This option is only available if the Data Object contains a Detail Visualization Section. Note: This option requires a <SetVisualizationObjectPropertiesConfiguration> section in your Request XML.
- Repeat Action – Repeat Action will close and re-open the Action Prompt, allowing you to run the action multiple times from a single menu click (Note: Repeat Action is only available if the Action displays a Prompt).
- Workflow – Workflow allows you to specify multiple Completion Actions along with Rules to specify which Completion Action from the workflow should be performed after the Action has completed.

Completion Action Document allows you to specify the path and name of the document that is to be downloaded/redirected to upon completion of the Action. If your Completion Action is Open URL, you should specify the full destination URL (e.g. http://www.google.com/images/srpr/logo3w.png). If your Completion Action is Open Document, the value of this field will be appended to the FTP Path field in the Data Source in order to get the full path to be downloaded via FTP from the back end system. If a URL is specified for Open Document, the target of the URL (an image, PDF, etc) will be downloaded from the Connector gateway and sent to the client. Completion

Action Document can also contain substitution values. See Appendix A: Specifying Messages and Substitution Values for more information about how to create a Completion Action Document. This value is only allowed if the Completion Action is Open Document or Open URL.

Completion Action Dashboard allows you to specify the Dashboard you wish to display on completion of the Action. See Chapter 14: Dashboards for more information on creating Dashboards

Data Object specifies which Data Object you want to display either a list or object of or perform an action against. This value is only allowed if the Completion Action is Open Object, Open List, or Open Action.

Rich Query specifies which Query you want to display on the list when using a rich client. This value is only allowed if the Completion Action is Open List or (under certain circumstances) Open Action (see Allowing Multiple Selections below for more information).

Mobile Query specifies which Query you want to display on the list when using a mobile client. This value is only allowed if the Completion Action is Open List or (under certain circumstances) Open Action (see Allowing Multiple Selections below for more information).

Rich Detail specifies which Detail you want to display on the object when using a rich client. This value is only allowed if the Completion Action is Open Object.

Mobile Detail specifies which Detail you want to display on the object when using a mobile client. This value is only allowed if the Completion Action is Open Object.

Hide Other Queries/Details specifies whether the runtime Query/Detail dropdown list will contain just the selected Rich/Mobile Query/Detail or all non-hidden Queries/Details. If Hide Other Queries/Details is selected, only the selected Rich/Mobile Query/Detail will be available to choose in the Query/Detail dropdown list.

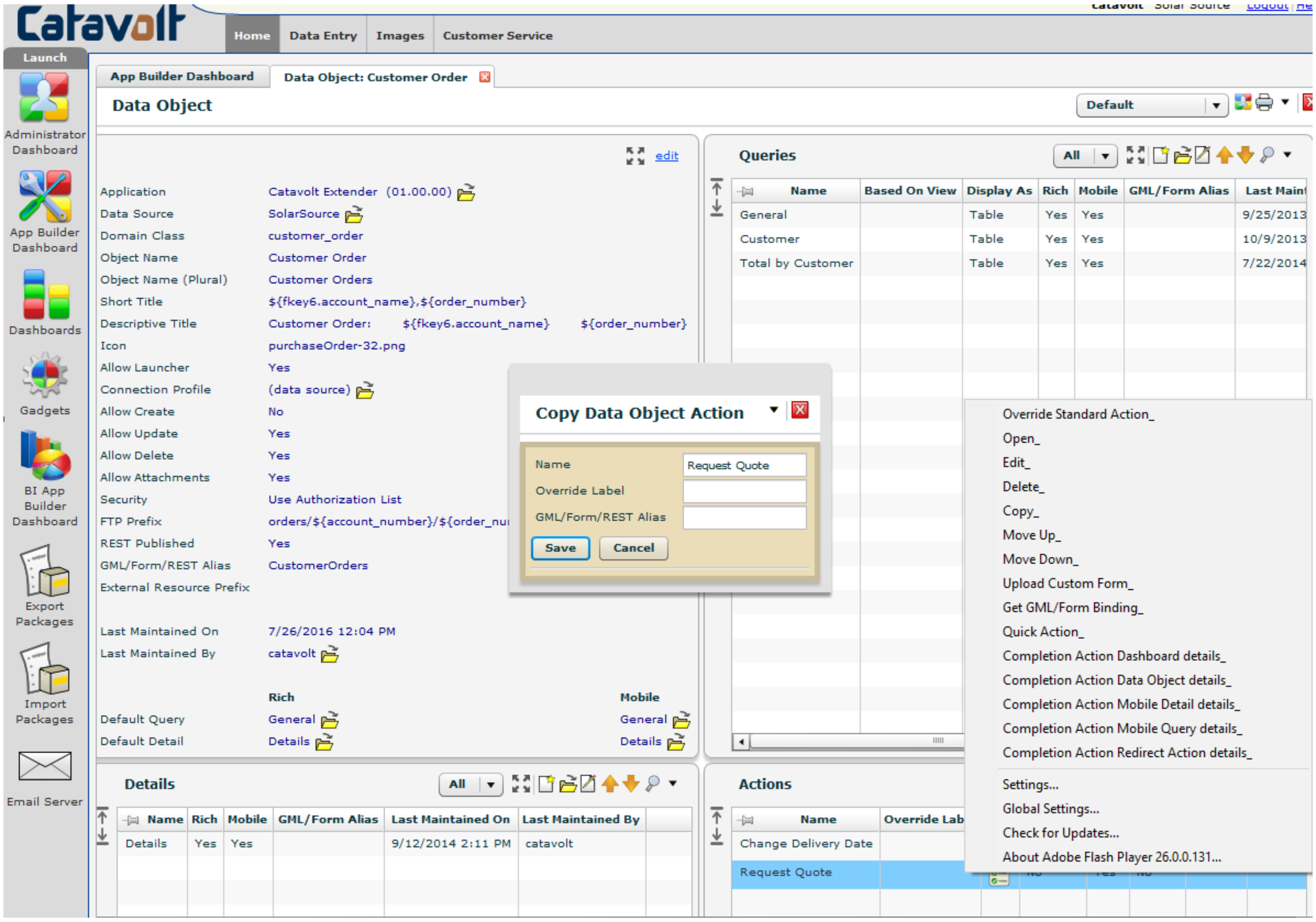
Action specifies which Action you want to perform on the selected data object. This value is only allowed if the Completion Action is Open Action.

Object Mode specifies to display the Detail in read-only or update mode when displaying an object. This value is only allowed if the Completion Action is Open Object.

Where Clause specifies extra criteria to use when querying an object. If the Completion Action is Open Object, Where Clause must be written to return a single record when it is performed (same as when creating a Dashboard Launcher for a Data Object). If the Completion Action is Open List, Where Clause can be optionally specified to further restrict the list of records returned. If the Completion Action is Open Action, Where Clause can be optionally specified to identify a single "selected record" to pass to the Action (NOTE: If you do not specify a Where Clause and the Action you have chosen is on the same Data Object, the current selected record (if any) will be passed to the Action). This value is only allowed if the Completion Action is Open Object, Open List, or Open Action.

Copying Actions

You may have instances where you need to make a copy of an existing Action. You can select the Copy menu option to accomplish this. When Copying an Action, you will be prompted to supply a new Name, Override Label, and GML/Form/REST Alias. An exact copy of this Action along with all of its components will be made.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_order
Object Name	Customer Order
Object Name (Plural)	Customer Orders
Short Title	${fkey6.account_name},${order_number}
Descriptive Title	Customer Order: ${fkey6.account_name} ${order_number}
Icon	purchaseOrder-32.png
Allow Launcher	Yes
Connection Profile	(data source)
Allow Create	No
Allow Update	Yes
Allow Delete	Yes
Allow Attachments	Yes
Security	Use Authorization List
FTP Prefix	orders/${account_number}/${order_nu
REST Published	Yes
GML/Form/REST Alias	CustomerOrders
External Resource Prefix	
Last Maintained On	7/26/2016 12:04 PM
Last Maintained By	catavolt

	Rich	Mobile
Default Query	General	General
Default Detail	Details	Details

Copy Data Object Action

Name	Request Quote
Override Label	
GML/Form/REST Alias	

Save Cancel

Queries

Name	Based On View	Display As	Rich	Mobile	GML/Form Alias	Last Maint
General		Table	Yes	Yes		9/25/2013
Customer		Table	Yes	Yes		10/9/2013
Total by Customer		Table	Yes	Yes		7/22/2014

Details

Name	Rich	Mobile	GML/Form Alias	Last Maintained On	Last Maintained By
Details	Yes	Yes		9/12/2014 2:11 PM	catavolt

Actions

Name	Override Lab
Change Delivery Date	
Request Quote	

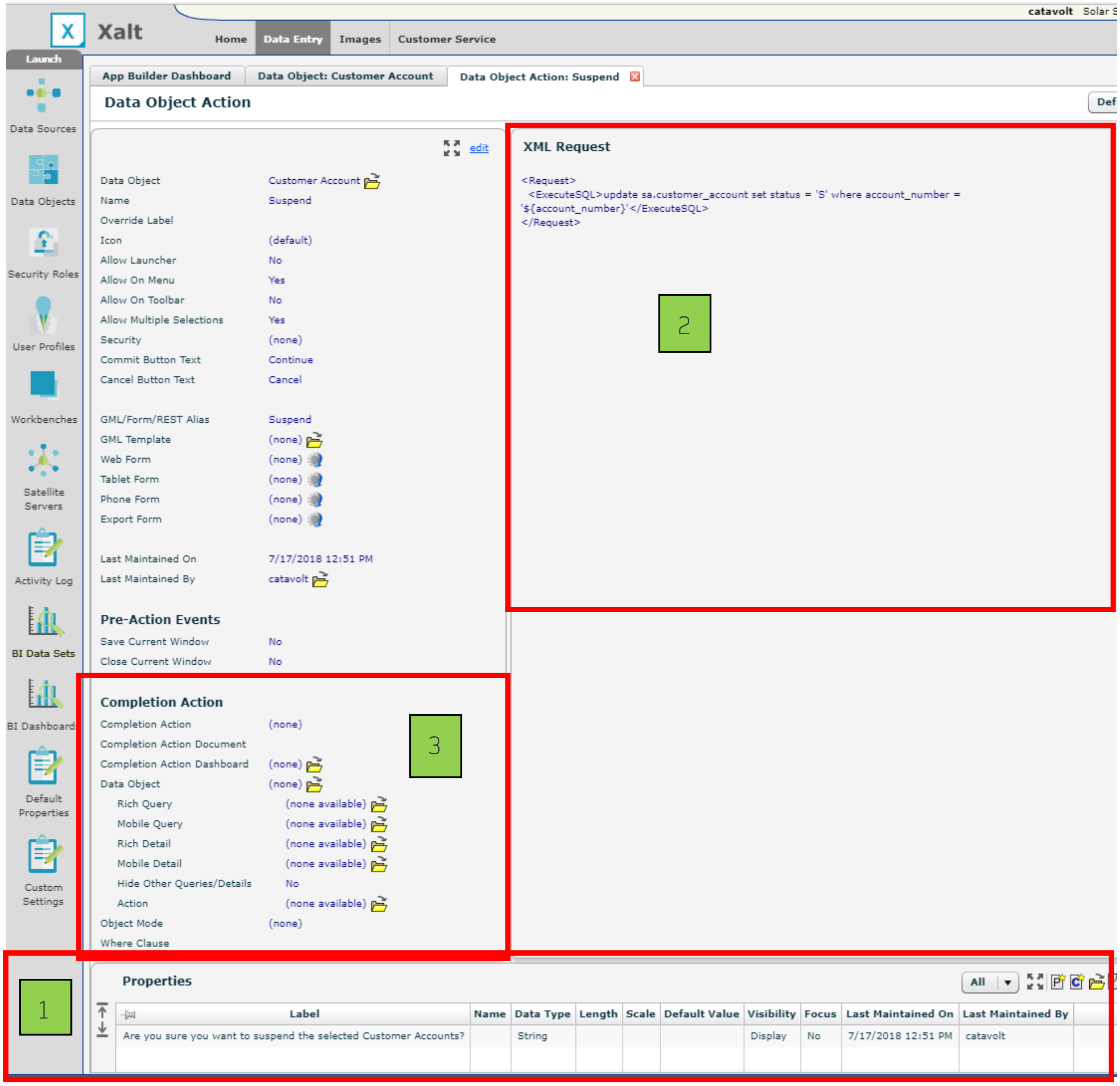
Figure 7: The Data Object Action copy prompt

Allowing Multiple Selections

By default, an Action can run against zero or one selected record only (depending on whether the Action requires property values from a selected record for Property Default Value, XML Request, Completion Action Document, etc). In enabling Actions to be performed against multiple selected records, we have enabled two different multi-select styles: Multi-Prompt and Single Prompt

Multi-Prompt

When selecting Multi-Prompt, the action will be performed over and over against each selected record. Basically, you are doing a Repeat Action against each selected record.



Data Object Action

Data Object	Customer Account
Name	Suspend
Override Label	
Icon	(default)
Allow Launcher	No
Allow On Menu	Yes
Allow On Toolbar	No
Allow Multiple Selections	Yes
Security	(none)
Commit Button Text	Continue
Cancel Button Text	Cancel
GML/Form/REST Alias	Suspend
GML Template	(none)
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	7/17/2018 12:51 PM
Last Maintained By	catavolt

Pre-Action Events

Save Current Window	No
Close Current Window	No

Completion Action

Completion Action	(none)
Completion Action Document	
Completion Action Dashboard	(none)
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Hide Other Queries/Details	No
Action	(none available)
Object Mode	(none)
Where Clause	

XML Request

```
<Request>
  <ExecuteSQL>update sa.customer_account set status = 'S' where account_number =
'${account_number}'</ExecuteSQL>
</Request>
```

Properties

Label	Name	Data Type	Length	Scale	Default Value	Visibility	Focus	Last Maintained On	Last Maintained By
Are you sure you want to suspend the selected Customer Accounts?		String				Display	No	7/17/2018 12:51 PM	catavolt

1) The Action Prompt will come up once for each selected record.
2) The XML Request will be performed once for each selected record. If a PreAction section is specified, it will also be performed once for each selected record.
3) The Completion Action will be performed ONCE only. If the Completion Action references fields from selected records (e.g. in the Where Clause), one selected record at random will be chosen to provide the selected values.

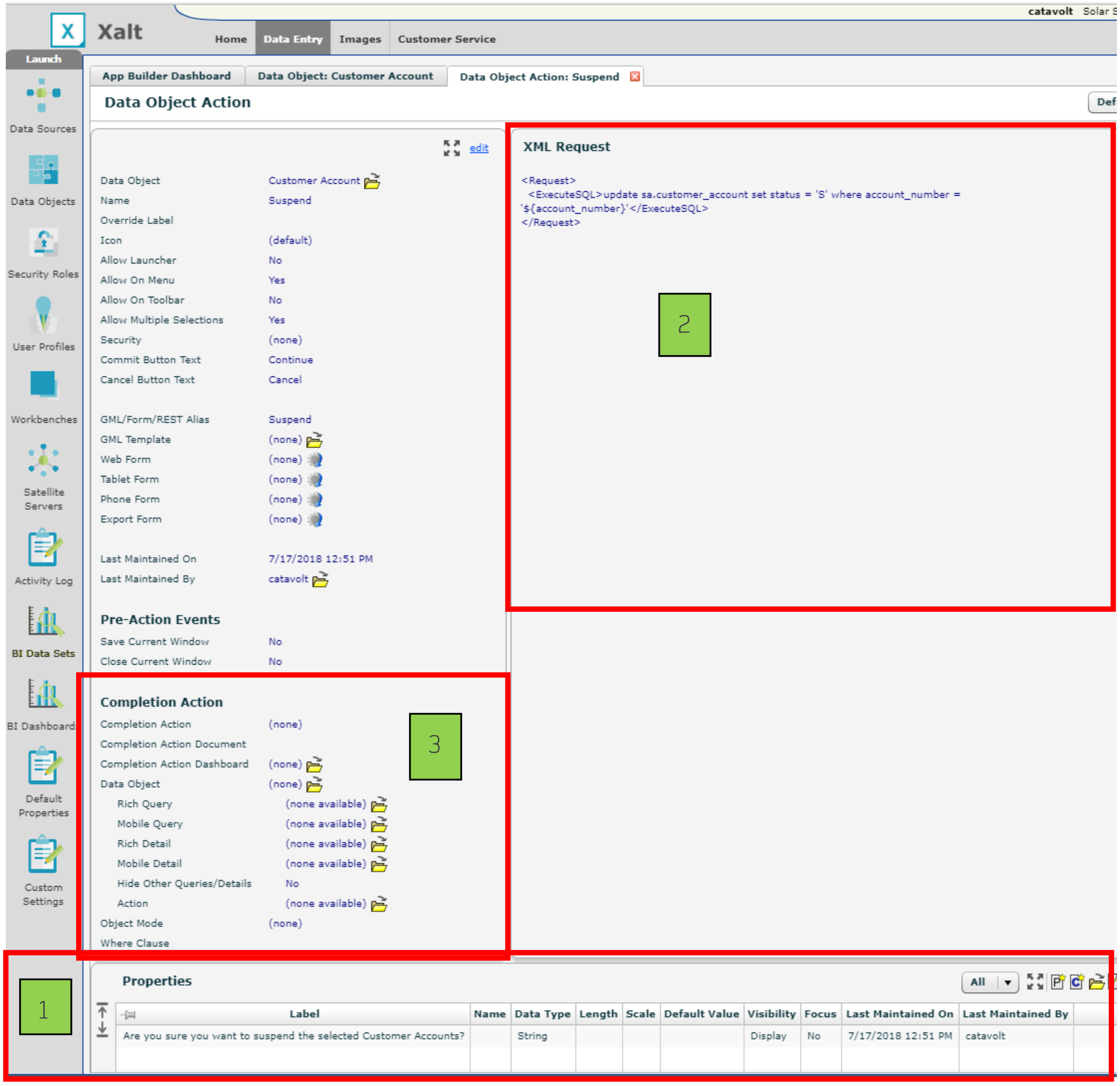
The Action flow when running a Multi-Prompt Action against multiple selections is as follows:

1) If a PreAction is defined, it is performed. If error messages are returned, processing stops and the error messages are returned. Any other messages returned are ignored and processing continues.
2) If a Prompt is defined, it is shown to the user.
3) The Request XML is performed. Any messages are shown to the end user (ignoring any Completion Actions) and processing is stopped.
4) Repeat 1-3 for each selected record.
5) If a Completion Action is defined, the Completion Action is performed.

If at any point in the above cycle, the user chooses Cancel or Back, processing stops and the prompt is cancelled. The Completion Action will not be performed.

Single Prompt

An Action running against multiple selected records with a single prompt performs slightly differently than one run with multiple prompts.



Data Object Action

Data Object	Customer Account
Name	Suspend
Override Label	
Icon	(default)
Allow Launcher	No
Allow On Menu	Yes
Allow On Toolbar	No
Allow Multiple Selections	Yes
Security	(none)
Commit Button Text	Continue
Cancel Button Text	Cancel
GML/Form/REST Alias	Suspend
GML Template	(none)
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	7/17/2018 12:51 PM
Last Maintained By	catavolt

Pre-Action Events

Save Current Window	No
Close Current Window	No

Completion Action

Completion Action	(none)
Completion Action Document	
Completion Action Dashboard	(none)
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Hide Other Queries/Details	No
Action	(none available)
Object Mode	(none)
Where Clause	

XML Request

```
<Request>
  <ExecuteSQL>update sa.customer_account set status = 'S' where account_number =
'${account_number}'</ExecuteSQL>
</Request>
```

Properties

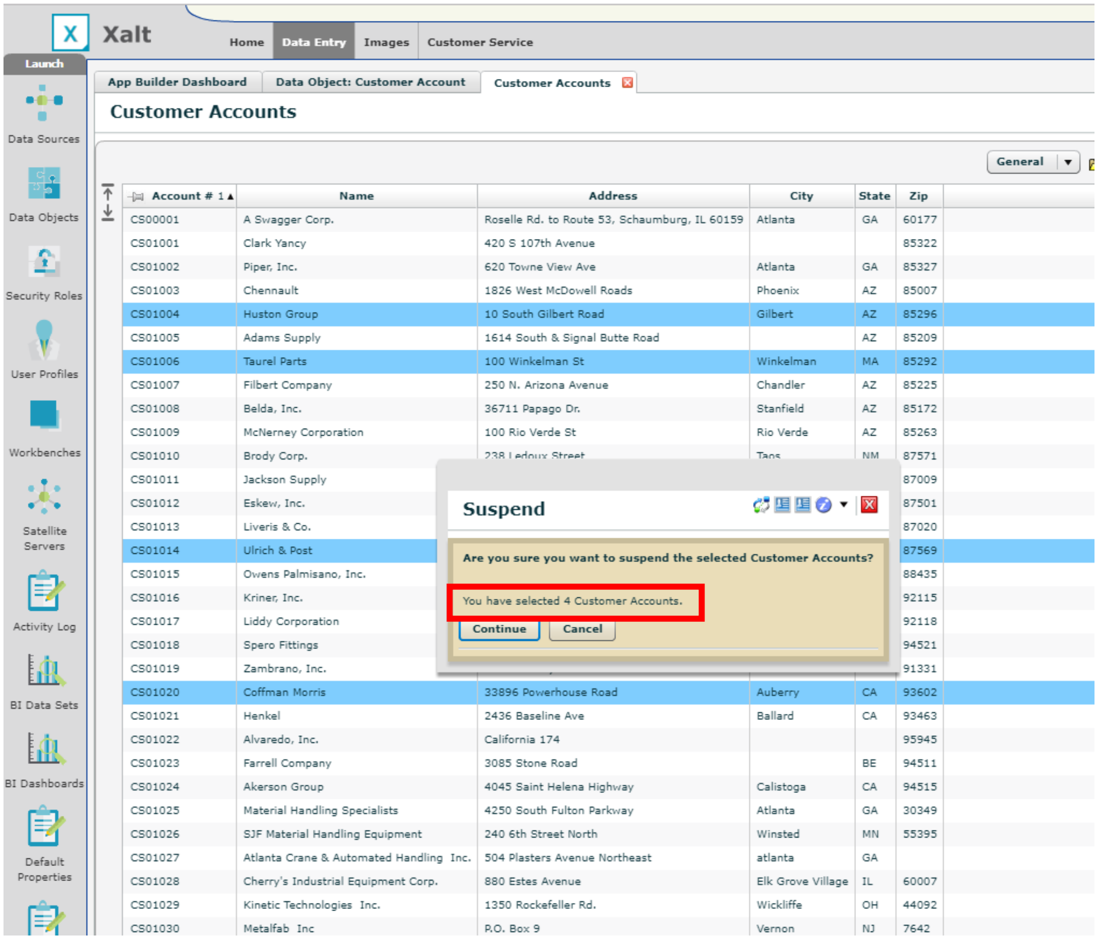
Label	Name	Data Type	Length	Scale	Default Value	Visibility	Focus	Last Maintained On	Last Maintained By
Are you sure you want to suspend the selected Customer Accounts?		String				Display	No	7/17/2018 12:51 PM	catavolt

1) The Action Prompt will come up ONCE only. If the prompt references fields from selected records (e.g. in the Default Value), one selected record at random will be chosen to provide the selected values.
2) The XML Request will be performed once against each selected record. If a PreAction section is specified, it will also be performed once against each selected record.
3) The Completion Action will be performed ONCE only. If the Completion Action references fields from selected records (e.g. in the Where Clause), one selected record at random will be chosen to provide the selected values.

The Action flow when running a Single Prompt Action against multiple selections is as follows:

1) If a PreAction is defined, it is performed against each selected record. All messages are collected during processing. If any error messages are returned, processing stops and the error messages are displayed. Any other messages returned are ignored and processing continues.
2) If a Prompt is defined, it is shown to the user.
3) The Request XML is performed for each selected record. Once all records have been processed, any messages are shown to the end user (ignoring any Completion Actions).
4) If a Completion Action is defined (and no messages were returned in 3) above), the Completion Action is performed.

Also note that if a prompt is displayed as part of the Single Prompt Action, a line at the bottom will be automatically added that indicates the number of selected records that the action will be performed against.



Customer Accounts

Account # 1 ▲	Name	Address	City	State	Zip
CS00001	A Swagger Corp.	Roselle Rd. to Route 53, Schaumburg, IL 60159	Atlanta	GA	60177
CS01001	Clark Yancy	420 S 107th Avenue			85322
CS01002	Piper, Inc.	620 Towne View Ave	Atlanta	GA	85327
CS01003	Chennault	1826 West McDowell Roads	Phoenix	AZ	85007
CS01004	Huston Group	10 South Gilbert Road	Gilbert	AZ	85296
CS01005	Adams Supply	1614 South & Signal Butte Road		AZ	85209
CS01006	Taurel Parts	100 Winkelman St	Winkelman	MA	85292
CS01007	Filbert Company	250 N. Arizona Avenue	Chandler	AZ	85225
CS01008	Belda, Inc.	36711 Papago Dr.	Stanfield	AZ	85172
CS01009	McNerney Corporation	100 Rio Verde St	Rio Verde	AZ	85263
CS01010	Brody Corp.	238 Ledoux Street	Taos	NM	87571
CS01011	Jackson Supply				87009
CS01012	Eskew, Inc.				87501
CS01013	Liveris & Co.				87020
CS01014	Ulrich & Post				87569
CS01015	Owens Palmisano, Inc.				88435
CS01016	Kriner, Inc.				92115
CS01017	Liddy Corporation				92118
CS01018	Spero Fittings				94521
CS01019	Zambrano, Inc.				91331
CS01020	Coffman Morris	33896 Powerhouse Road	Auberry	CA	93602
CS01021	Henkel	2436 Baseline Ave	Ballard	CA	93463
CS01022	Alvaredo, Inc.	California 174			95945
CS01023	Farrell Company	3085 Stone Road		BE	94511
CS01024	Akerson Group	4045 Saint Helena Highway	Calistoga	CA	94515
CS01025	Material Handling Specialists	4250 South Fulton Parkway	Atlanta	GA	30349
CS01026	SJF Material Handling Equipment	240 6th Street North	Winsted	MN	55395
CS01027	Atlanta Crane & Automated Handling Inc.	504 Plasters Avenue Northeast	atlanta	GA	
CS01028	Cherry's Industrial Equipment Corp.	880 Estes Avenue	Elk Grove Village	IL	60007
CS01029	Kinetic Technologies Inc.	1350 Rockefeller Rd.	Wickliffe	OH	44092
CS01030	Metalfab Inc	P.O. Box 9	Vernon	NJ	7642

Suspend

Are you sure you want to suspend the selected Customer Accounts?

You have selected 4 Customer Accounts.

Continue Cancel

Multi-Select Actions with Completion Actions

The Open Action Completion Action can be used for one Action to call a second Action. When calling an Action on the same Data Object and no Where Clause is specified, the selected record is passed from the original Action to the Open Action Completion Action. If the Open Action is on a different Data Object than the original Action, no selected record is passed to the new Action.

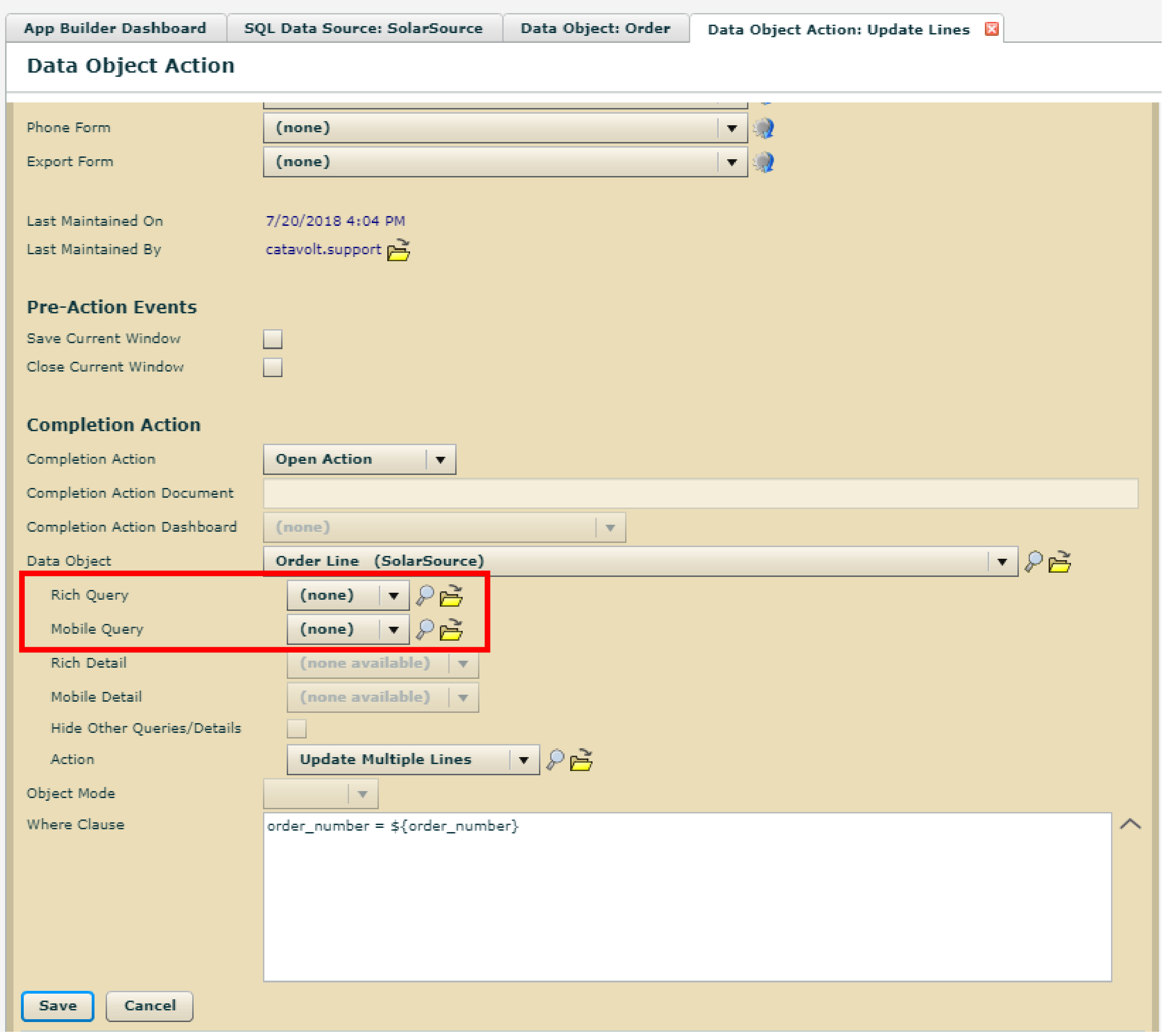
This standard behavior affects multi-select actions as follows:

- If a multi-select action calls another multi-select action on the same Data Object, the (multiple) selected records will be passed to the new Action.
- If a single-select action calls a multi-select action on the same Data Object, the (single) selected record will be passed to the new Action
- If a multi-select action calls a single-select action on the same Data Object, the selected records will be passed to the new Action. If the original Action was called with more than one selected record, an error will occur.

You also have the option to override the selected records that are passed to the Open Action Completion Action by using the Where Clause. Xalt will perform a Query using the Where Clause to return a list of records to pass as "selected records" to the Open Action Completion Action. This affects multi-select actions as follows:

- If the Where Clause returns multiple records and the Open Action supports multi-select, the entire list of records will be passed as selected records. The list of selected records will be ordered by primary key. Note that a max of 1000 selected records will be passed to the Open Action Completion Action.
- If the Where Clause returns multiple records and the Open Action does not support multi-select, an error will occur.

If the Open Action Completion Action is a Multi-Prompt Action, the developer may want to change the order of the selected records from the primary key (as a prompt is displayed for each selected record in order). In order to support this, we are allowing the user to specify a Rich/Mobile Query in addition to the Where Clause if the Open Action Completion Action is Yes (Multi-Prompt):


App Builder Dashboard
SQL Data Source: SolarSource
Data Object: Order
Data Object Action: Update Lines
Data Object Action

Phone Form: (none)
Export Form: (none)
Last Maintained On: 7/20/2018 4:04 PM
Last Maintained By: catavolt.support

Pre-Action Events
Save Current Window: □
Close Current Window: □

Completion Action
Completion Action: Open Action
Completion Action Document:
Completion Action Dashboard: (none)
Data Object: Order Line (SolarSource)
Rich Query: (none)
Mobile Query: (none)
Rich Detail: (none available)
Mobile Detail: (none available)
Hide Other Queries/Details: □
Action: Update Multiple Lines
Object Mode:
Where Clause: order_number = ${order_number}

The following rules apply:

1) The Open Action MUST be Allow Multiple Selections = Yes (Multi-Prompt)
2) A Where Clause MUST be specified
3) Query is available if 1) and 2) are satisfied – choices are (none) or an existing Query
4) If a Query is specified, the Sort Ordering from the Query will be used to order the selected records
5) If the Query contains a Where Clause, it will be added to the Where Clause in the Completion Action when querying selected records.

The above behavior is also available when calling Open Action via a Workflow Completion Action.

XML Request

XML Request specifies the work that is to be performed on the back end system when the action is performed. Xalt | Mobility allows a number of different types of XML Requests depending on the Data Source type. Note that you may include multiple actions in your XML Request. You can even mix and match action types inside a single request (assuming your Data Source supports each type). Your XML Request can also contain Substitution Values. See Appendix A: Specifying Messages and Substitution Values for more information about how to use Substitution Values in an XML Request.

Direct SQL Requests

You can specify direct SQL actions if your Data Source type is IBM i Direct or SQL Direct. An example of the syntax is:

```
<Request>
 <ExecuteSQL>
  UPDATE service_orders SET customer_quote = '${NEW_QUOTE}' where order_number = '${order_number}'
 </ExecuteSQL>
</Request>
```

You should specify the SQL statement using syntax understood by your back end system.

JDBC Stored Procedures

You can call a JDBC stored procedure if your Data Source type is IBM i Direct or SQL Direct. An example of the syntax is:

```
<Request>
 <ExecuteJDBCProgram>
  <Name>update_product_price</Name>
  <ParameterList>
   <Parameter type='String' length='10' direction='IN'>
    <Value><![CDATA[${PRODUCT_NUMBER}]]></Value>
   </Parameter>
   <Parameter type='Decimal' length='8' scale='2' direction='IN'>
    <Value><![CDATA[${NEW_PRICE}]]></Value>
   </Parameter>
  </ParameterList>
 </ExecuteJDBCProgram>
</Request>
```

Name specifies the name of the JDBC stored procedure to call

ParameterList specifies the parameters for the stored procedure call.

Parameter specifies a single parameter to the program. It contains the following values:

Type specifies the type of parameter. The list below shows the allowable values followed by the corresponding SQL parameter types as registered with the stored procedure call:

- Boolean – SQL Type BOOLEAN (16)
- Date – SQL Type DATE (91)
- Decimal – SQL Type DECIMAL (3)
- Integer – SQL Type INTEGER (4)
- Large Binary – SQL Type VARBINARY (-3)
- Whole Number – SQL Type BIGINT (-5)
- String – SQL Type VARCHAR (12)
- Text – SQL Type VARCHAR (12)
- Time – SQL Type TIME (92)
- Timestamp – SQL Type TIMESTAMP (93)

Length specifies the length of the Parameter.

Scale specifies the number of decimals of the Parameter. This value is only required for types that support decimals.

Direction specifies whether this parameter is input, output, or both. The allowable values are:

- IN – A value will be passed in as input to the program
- OUT – A value will be returned from the program as output.
- INOUT – A value will be passed into the program and returned from the program.

Note that you can currently specify a parameter as OUT or INOUT, however there is no mechanism to retrieve or show the value when the program returns except for return XML as specified below.

Value specifies the value to be passed into the stored procedure call. This value is only specified if the Direction is IN or INOUT.

AS/400 Direct Program Calls

You can directly call an AS/400 program if your Data Source type is IBM i Direct. An example of the syntax is:

```
<Request>
 <ExecuteAS400Program>
  <Name>UPDPRDPRC</Name>
  <ParameterList>
   <Parameter type=’String’ length=’10’ direction=’IN’>
    <Value><![CDATA[${PRODUCT_NUMBER}]]></Value>
   </Parameter>
   <Parameter type=’Decimal’ length=’8’ scale=’2’ direction=’IN’>
    <Value><![CDATA[${NEW_PRICE}]]></Value>
   </Parameter>
  </ParameterList>
 </ExecuteAS400Program>
</Request>
```

Name specifies the name of the AS/400 program to call.

All other values specified are the same as for JDBC Stored Procedures. Please see the section above for more information.

HTTP Web Service Calls

You can call an HTTP Web Service if your Data Source type is IBM i Direct or SQL Direct. An example of the syntax is:

```
<Request>
 <ExecuteHTTPAction>
  <URL>http://kabr.mybiz.net:8080</URL>
  <Method>POST</Method>
  <ContentType>application/soap+xml; charset=utf-8</ContentType>
  <SetHeaderField name='Keep-Alive'>300</SetHeaderField>
  <SetHeaderField name='JSESSIONID'>${S_MYSESSIONID}</SetHeaderField>
  <Content><![CDATA[<?xml version="1.0" encoding="utf-8"?>
          <soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
                   xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
           <soap12:Body>
            <CreateContact xmlns="http://kabr.mybiz.net/">
             <inUser>${P_USER}</inUser>
            </CreateContact>
           </soap12:Body>
          </soap12:Envelope>]]></Content>
 </ExecuteHTTPAction>
</Request>
```

URL specifies the URL to connect to make the call

Method specifies the method for the URL request. This value is typically one of the following: GET, POST, HEAD, OPTIONS, PUT, DELETE, or TRACE.

ContentType specifies the value for the "content-type" property on the HTTP call.

SetHeaderField allows you to set one or more additional properties on the HTTP call.

Content specifies the body content of the HTTP Call.

Ignoring the Response

When performing an <ExecuteHttpAction> request, we will always process the response, expecting it to be XML or JSON. If your Web Server is returning data in a format other than XML or JSON, or you do not care about the response and would like to skip processing, add processResponse='false' to the <ExecuteHttpAction> tag. For example:

```
<Request>
  <ExecuteHTTPAction processResponse='false'>
...
  </ExecuteHTTPAction>

</Request>
```

JSON Content

Allowing JSON content in an HTTPActionCall presents a different issue. When dealing with XML, you can use a CDATA section in cases where you need to include user-provided values that may contain XML control characters. Unfortunately, there is no similar provision for JSON. Here we have JSON Content in a Web Service call:

```
<Content><![CDATA[
       {"Class":"SDAMobileComment",
              "Description":"${W_DESCRIPTION}",
              "Name":"${W_NAME}", "MobileCommentTag@odata.bind":"${CURRENT_SYSTEM_URL}/Objects('${Id}')
       }]]>
</Content>
```

The user is prompted to enter a Name and Description. If either of these values contain a " character, the JSON will fail. Since Name and Description are provided by the user, they may or may not need to be escaped for JSON control characters. We have implemented a <Declare> section in the XML Request to accomplish this:

```
<Declare>
    <Variable name='W_NAME'>
      <Value format='JSON'><![CDATA[${P_NAME}]]></Value>
    </Variable>
    <Variable name='W_DESCRIPTION'>
      <Value format='JSON'><![CDATA[${P_DESCRIPTION}]]></Value>
    </Variable>
  </Declare>
```

In this section, we define a variable called W_NAME. the value of the variable is ${P_NAME}, the value the user entered on the prompt. The format='JSON' attribute specifies that this variable is used as JSON data. Xalt will take care of escaping any JSON control characters when setting the value. You can now use ${W_NAME} in the XML Request to get the JSON-escaped version of ${P_NAME}

Infor System-Link Requests

If the Data Source type is Infor System-Link, the only type of XML Request allowed is Infor System-Link Requests. When you create an Action for an Infor System-Link Data Source, the XML Request is defaulted to:

<Request sessionHandle='&sessionHandle&' workHandle='*new' broker='EJB' maxIdle='0'></Request>

You should leave this part as-is and insert your actions inside the Request tag:

```
<Request sessionHandle=’&sessionHandle&’ workHandle=’*new’ broker=’EJB’ maxIdle=’0’>
 <Update name=’updateObject_Buyer’ domainClass=’com.mapics.pm.Buyer’>
  <DomainEntity>
   <Key>
    <Property path=’buyer’><Value><![CDATA[${buyer}]]></Value></Property>
   </Key>
   <Property path=’buyerName’><Value><![CDATA[${NEW_NAME}]]></Value></Property>
  </DomainEntity>
 </Update>
</Request>
```

Please refer to the Infor’s Guide To XA System-Link for more information on formatting Infor System-Link requests.

Setting Session Properties

A Session Property is a property that is only available for the current Hexagon Session. A good way to think about it is that a Session Property is like a User Property that is reset every time a user logs into the system.   You can directly set a Session Property as part of an XML Request. An example of the syntax is:

```
<Request>
 <SetSessionProperty>
    <SessionProperty name=’S_USERID’><Value>HEB</Value></SessionProperty>
    <SessionProperty name=’S_WHID’ scope='local'><Value>${P_NEW_WHS}</Value></SessionProperty>
  </SetSessionProperty>
</Request>
```

Name specifies the name of the Session Property that you wish to set. You should ensure that this name does not match the name of other Substitution Values or properties from the Data Object to keep any name clashes from occurring. We recommend making the name all uppercase (case is sensitive) or otherwise doing something to make the name appear unique.

Scope specifies the scope (global/local) of the Session Property that you wish to set. You can set this to global (the value is available everywhere) or local (the value is only available for the current window and any child windows). The default value if not specified is global. See Appendix A: Specifying Messages and Substitution Values for more information about the differences between Global vs. Local Session Properties.

Value specifies the value to be set. This can contain constants or substitution values. See Appendix A: Specifying Messages and Substitution Values for more information about using Session Properties.

Setting User Profile Properties

A User Profile Property is a property that is available for the current user. You can directly add/update/delete a User Property as part of an XML Request. An example of the syntax is:

```
<Request>
 <SetUserProperty>
  <UserProperty name='U_USERID'><Value>HEB</Value></UserProperty>
  <UserProperty name='U_DEPTID'><Value>AA1A</Value></UserProperty>
  <RemoveUserProperty name='U_OLDUSERID'/>
 </SetUserProperty>
</Request>
```

Name specifies the name of the User Property that you wish to set. You should ensure that this name does not match the name of other Substitution Values or properties from the Data Object to keep any name clashes from occurring. We recommend making the name all uppercase (case is sensitive) or otherwise doing something to make the name appear unique.

Value specifies the value to be set. This can contain constants or substitution values. If the current User Profile already has a User Profile Property with this name, the value will be updated, otherwise a new User Profile Property will be created. See Appendix A: Specifying Messages and Substitution Values for more information about using Session Properties.

Note that the <UserProperty> tag is used to create or update User Profile Properties, while the <RemoveUserProperty> tag is used to delete existing User Profile Properties.

Setting Default Properties

A Default Property is a property that is available for all users. You can directly add/update/delete a Default Property as part of an XML Request. An example of the syntax is:

```
<Request>
 <SetDefaultProperty>
  <DefaultProperty name='U_USERID'><Value>HEB</Value></DefaultProperty>
  <DefaultProperty name='U_DEPTID'><Value>AA1A</Value></DefaultProperty>
  <RemoveDefaultProperty name='U_OLDUSERID'/>
 </SetDefaultProperty>
</Request>
```

Name specifies the name of the Default Property that you wish to set. You should ensure that this name does not match the name of other Substitution Values or properties from the Data Object to keep any name clashes from occurring. We recommend making the name all uppercase (case is sensitive) or otherwise doing something to make the name appear unique.

Value specifies the value to be set. This can contain constants or substitution values. If a Default Property already exists with this name, the value will be updated, otherwise a new Default Property will be created. See Appendix A: Specifying Messages and Substitution Values for more information about using Default Properties.

Note that the <DefaultProperty> tag is used to create or update Default Properties, while the <RemoveDefaultProperty> tag is used to delete existing Default Properties.

Setting EDE Configuration

An EDE Configuration is used in conjunction with the **Open EDE View** Completion Action. SetEDEConfiguration specifies the metadata and data web service calls necessary to display an EDE View. An example of the syntax is:

```
<Request>
<SetEDEConfiguration>
  <Metadata>
    <ExecuteHTTPAction>
      <Method>GET</Method>
<URL>${CURRENT_SYSTEM_URL}/${SELECTED_ODATA_OBJECT_ID}?$format=application/json;odata.metadata=full
</URL>
    <SetHeaderField name='prefer'><![CDATA[odata.include-annotations=*]]></SetHeaderField>
    <Content></Content>
  </ExecuteHTTPAction>
  </Metadata>
  <Data>
    <ExecuteHTTPAction>
      <Method>POST</Method>
<URL>${CURRENT_SYSTEM_URL}/${SELECTED_ODATA_OBJECT_ID}/Com.Ingr.SiApi.V2.ExecuteEngineeringDataEdit
or</URL>
      <ContentType>application/json</ContentType>
<Content><![CDATA[{objectHierarchy:{Name:"AAA",Path:"null",ResourceIds:["HighestPlantGroups('9999')/Intermediat
ePlantGroupHighestPlantGroup('5555')/LowestPlantGroupIntermediatePlantGroup('1111')"]}}]]></Content>
  </ExecuteHTTPAction>
  </Data>
</SetEDEConfiguration>
</Request>
```

Metadata specifies the ExecuteHttpAction that should be performed to retrieve the metadata necessary to display the EDE View.

Data specifies the ExecuteHttpAction that should be performed to retrieve the data necessary to populate the EDE View.

Display Visualization Section

Retrieving Graphic ID

Once the Visualization Section is displayed, Custom actions in Extender need to be able to be performed against it. The user can select an object inside the Visualization Section and retrieve the Graphic ID for use in performing an Action. Extender will add support for the following Substitution Values for use by Actions run while inside the Visualization Section:

${SELECTED_GRAPHIC_ID} – This will return a String that represents the selected object's Model/Moniker ID (e.g. @a=0027!!20014##298465568039508828)

${SELECTED_GRAPHIC_ID@URLENCODE} -- This will return a String that represents the selected object's Model/Moniker ID. The value returned will be URL encoded so that it can be seamlessly embedded in another URL (e.g. %40a%3D0027%21%2120014%23%23298465568039508828)

Displaying Object Properties

Since there is no common API for retrieving and displaying information for a selected object inside a Visualization Section, this will be done via a custom action. Inside the Visualization Section, you may specify a **Default Action** that represents a Custom action built to retrieve this information. When the user selects an object in the Visualization Section, the client will call the Default Action as it would any other custom action (Note that this action will most likely use the ${SELECTED_GRAPHIC_ID} values as outlined above to retrieve a Moniker/Model ID).

There are currently 2 methods PPM uses to retrieve properties: a standard Object Details call, and a specific GetObjectProperties() call that returns a curated list of properties.

Standard Object Details Call

In this scenario (used by most current PPM products), the back-end system does not have a specific call that you can pass a Graphic ID into and retrieve a specific Property list to display to the user. Instead, you would make a normal ODATA object call and display the details. For example, this call is made:

GET https://hexlive19-spel.hexagonppm.com/WebApi/SEL/V2/Sites('SITE001')/Plants('PLANT002')/Motors('06CFEBD38E2341BAA9ADB2516F68B8EC')

This is the return JSON value:

{
"@odata.context":"https://hexlive19-spel.hexagonppm.com/WebApi/SEL/V2/$metadata#Sites('SITE001')/Plants('PLANT002')/Motors/$entity","@odata.type":"#Com.Ingr.SEL.V2.SITE001.PLANT002.Motor","Id":"06CFEBD38E2341BAA9ADB2516F68B8EC","ConcurrencyToken":"06CFEBD38E2341BAA9ADB2516F68B8EC","MotorDesignLetter":"","TeStallTimeSiValue":null,"TeStallTime":"","MinimumRequiredPowerSiValue":null,"MinimumRequiredPower":"","MotorEfficiencyClass":"","LRCToFLARatio":5.2,"MotorControllerType":"","VFDRequiredFlag":"","RunUpTimeSiValue":null,"RunUpTime":"","AsynchronousSpeedSiValue":null,"AsynchronousSpeed":"","NumberOfPoles":"4","VibrationMonitoringFlag":"","SpeedControlRange":"","SpeedControlFlag":"","SignalSpeedReq":"","SignalFwdRevReq":"","LRCCodeLetterFlag":"","LRCCodeLetter":"","FwdRevControlFlag":"","JBCableEntrySizeSiValue":null,"JBCableEntrySize":"","RestartGroup":"","RestartFlag":"","TcStallTimeSiValue":null,"TcStallTime":"","WindingTempCutOffFlag":"","MotorRatedPower":"1 hp","SP_ID":"06CFEBD38E2341BAA9ADB2516F68B8EC","ConformityToStandard":"","SynchronousSpeedSiValue":188.495559215388,"SynchronousSpeed":"1800.0 rpm","PowerFactor75Load":0.83,"ParticularReactiveLoadSiValue":null,"ParticularReactiveLoad":"","ParticularPowerSiValue":null,"ParticularPower":"","ParticularApparentLoadSiValue":null,"ParticularApparentLoad":"","ParticularActiveLoadSiValue":null,"ParticularActiveLoad":"","PowerFactorHalfLoad":0.81,"OperatingFactor":0.8,"RequiredStarterType":"","MotorInsulationClass":"","TemperatureRise":"","KVAtoHPRatio":null,"KVAperHPFactor":null,"MotorNo":4,"StartingCurrentSiValue":6.2,"StartingCurrent":"6.2 A","ProcessOperationalRequirement":"","RemoteControlStationReq":"","SignalOutOfServiceReq":"","SignalFaultReq":"","SignalStandByReq":"","SignalRemoteReq":"","SignalStatusControlReq":"","EmergencyStopControlFlag":"","LocalControlStationReq":"","EfficiencyFullLoad":0.9,"PowerFactorFullLoad":0.85,"StopControlFlag":"","StartControlFlag":"","PowerFactorAtStarting":0.4,"AverageActiveLoadSiValue":530.0,"AverageActiveLoad":"0.53

kW","RatedActiveLoadSiValue":830.0,"RatedActiveLoad":"0.83
kW","RatedApparentLoadSiValue":970.0,"RatedApparentLoad":"0.97
kVA","RatedReactiveLoadSiValue":510.0,"RatedReactiveLoad":"0.51 kVAR",

…

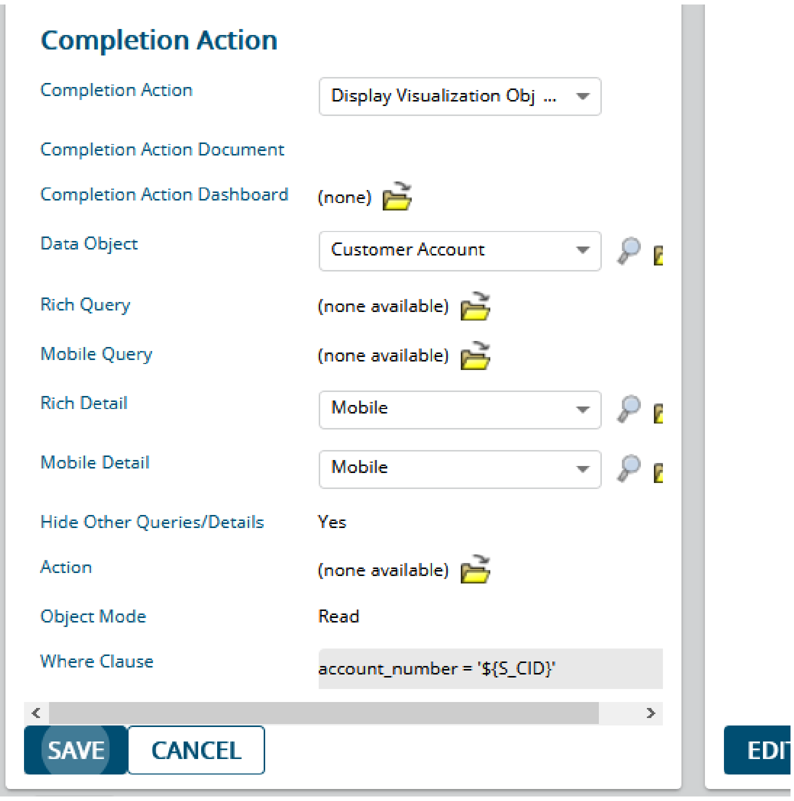
"SP_PlantGroupID":"500D6A67A1294AEB87CE20DB1C61BA26","SP_PlantItemGroupID":"","SP_SameAsUID":"","SP_SPIID
":"","SP_EquipID":"06CFEBD38E2341BAA9ADB2516F68B8EC","SP_CaseID":"06B8E5CC06E743AEBC1B78FA7971662D
","StructureType":"PlantItem"
}

There are 2 problems with this approach:

1) The base response returns way too many data properties. There needs to be some way to filter the list of properties returned.
2) The response only returns property name and value. There is no "DisplayName" equivalent that will show a user-appropriate label. While most property names are similar to what an appropriate "DisplayName" would be, there are many that are not.

Extender already handles these issues when displaying details for a single record (you create a Detail Property Section to define what properties and labels to show). We use this approach to resolve these issues for this scenario.

In this scenario, if the Data Object contains a Detail Visualization Section, a new Completion Action of **Display Visualization Object Properties** will be available. You specify the Data Object, Detail, and Where Clause to display as you would with a normal Open Object Completion Action:



Completion Action

Completion Action	Display Visualization Obj ...
Completion Action Document	
Completion Action Dashboard	(none)
Data Object	Customer Account
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	Mobile
Mobile Detail	Mobile
Hide Other Queries/Details	Yes
Action	(none available)
Object Mode	Read
Where Clause	account_number = '${S_CID}'

The XML Request can be used (if needed) to gather necessary information from the Graphic ID and set Session Properties for the result that can be used in the Where Clause. For example:

```
<Request>
  <ExecuteHTTPAction>
    <Method>GET</Method>
    <URL><![CDATA[${CURRENT_SYSTEM_URL}/Sites('SITE001')/Plants('PLANT002')/
PlantItems('${SELECTED_GRAPHIC_ID}')]]></URL>
    <ContentType>application/json</ContentType>
    <Content><![CDATA[]]></Content>
    <Return>
      <SetSessionProperty name='S_CID'>
        <JSONPath><![CDATA[/value/Id]]></JSONPath>
      </SetSessionProperty>
    </Return>
  </ExecuteHTTPAction>
</Request>
```

We also allow Workflow Completion Actions for this type of scenario. You can use the Rules section to choose the desired Data Object / Detail to open based on Session Properties built in the XML Request.

Extender will retrieve the first Detail Property Section in the specified detail and send the properties, labels, and values to the Visualization Section, which will show this information to the user inside the Visualization Section (as opposed to opening a new Extender window).

Specific GetObjectProperties() call

In this scenario, the PPM back-end system has a specific call that you can pass a Graphic ID into and retrieve a specific ad-hoc Property list to display to the user. For example, the following call:

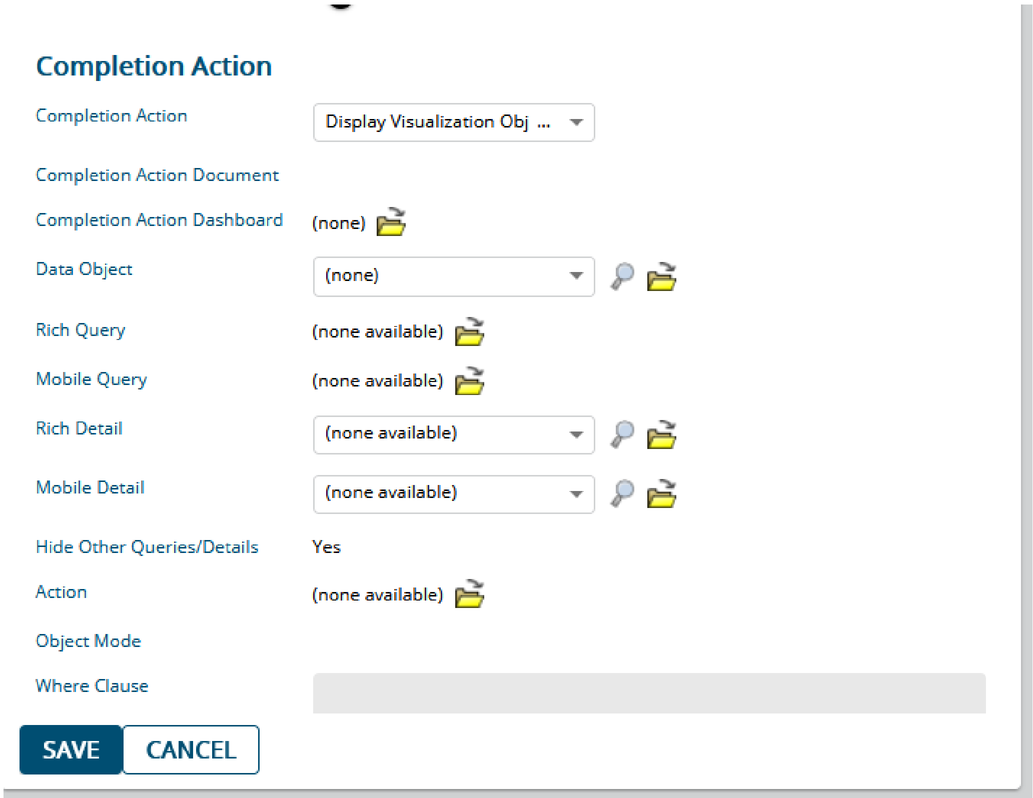
https://sdxsrvr.hexagonppm.com/SDxServer/api/v2/SMARTREV/GetObjectPropertiesForRepresentation?GraphicOID='%40a%3D0027%21%2120014%23%23297616833367711773'&FileOBID='6FKE000A'

returns the following JSON:

```
{"@odata.context":"<url>/$metadata#Collection(Intergraph.SPF.Server.API.ClientSupport.Types.SPFProperty)",
"value":[
    {"DisplayName":"Quality status","Name":"SPFNQualityStatus","Value":"Unknown"},
    {"DisplayName":"Transfer status","Name":"SPFNTransferStatus","Value":"None"},
…
    {"DisplayName":"Creation Date","Name":"CreationDate","Value":"2019/05/30-14:33:10:355"}

]}
```

As above, if the Data Object contains a Detail Visualization Section, a new Completion Action of **Display Visualization Object Properties** will be available.



Completion Action

Completion Action	Display Visualization Obj ...
Completion Action Document	
Completion Action Dashboard	(none)
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Hide Other Queries/Details	Yes
Action	(none available)
Object Mode	
Where Clause	

In this case, the app designer would specify a Data Object of (none). Instead, they will supply the information needed to retrieve the Object Properties in the XML Request in the following format:

```
<Request>
<SetVisualizationObjectPropertiesConfiguration>
  <Data>
    <ExecuteHTTPAction>
      <Method>POST</Method>
      <URL> <![CDATA[${CURRENT_SYSTEM_URL-1}/SMARTREV/GetObjectPropertiesForRepresentation?
GraphicOID='${SELECTED_GRAPHIC_ID@URLENCODE}'&FileOBID='${S_LVD_SELECT_FILE_ID}']]></URL>
      <ContentType>application/json</ContentType>
      <Content><![CDATA[]]></Content>
     <SetHeaderField name='SPFCreateConfigUID'>${S_LVD_CONFIGUID}</SetHeaderField>
    </ExecuteHTTPAction>
  </Data>
</SetVisualizationObjectPropertiesConfiguration>
</Request>
```

The <ExecuteHTTPAction> inside the <Data> section will make the call that returns the JSON Object with the properties data. We consider “standard” format to be the following:

```
{"@odata.context":"<url>/$metadata#Collection(Intergraph.SPF.Server.API.ClientSupport.Types.SPFProperty)",
"value":[
   {"DisplayName":"Quality status","Name":"SPFNQualityStatus","Value":"Unknown"},
   {"DisplayName":"Transfer status","Name":"SPFNTransferStatus","Value":"None"},
....
   {"DisplayName":"Description","Name":"Description","Value":"Brew Kettle"}

]}
```

The JSON Object will return an array of properties in the “value” top-level entry. Each array entry will contain a “Name” property that will map to the property name. There will also be a “DisplayName” property that will map to the property label and a “Value” property that will map to the property value.

As with the previous scenario, Extender will retrieve the information from the server call and send the properties, labels, and values to the Visualization Section, which will show this information to the user inside the Visualization Section (as opposed to opening a new Extender window).

URL Encoding Values

You have the ability to URL Encode values as part of an XML Request. This can be useful where user-entered values or values returned from an HTTP call need to be used as part of the URL in a subsequent HTTP call.

The format for URL encoding a value inside an XML Request is:

```
<PerformUrlEncode>
  <Value><![CDATA[]]></Value>
  <Encoding>UTF-8</Encoding>
  <SetSessionProperty name='SESSION_PROPERTY_NAME' />
  <SetUserProperty name='USER_PROPERTY_NAME' />
  <SetDefaultProperty name='DEFAULT_PROPERTY_NAME' />
</PerformUrlEncode>
```

Value specifies the value that needs to be URL encoded. This tag is required.

Encoding specifies the encoding character set to use. This tag is optional and will default to “UTF-8” if not specified.

SetSessionProperty specifies the name of the Session Property to receive the encoded value. You should ensure that this name does not match the name of other Substitution Values or properties from the Data Object to keep any name clashes from occurring. We recommend making the name all uppercase (case is sensitive) or otherwise doing something to make the name appear unique. This tag is optional

SetUserProperty specifies the name of the User Property to receive the encoded value. You should ensure that this name does not match the name of other Substitution Values or properties from the Data Object to keep any name clashes from occurring. We recommend making the name all uppercase (case is sensitive) or otherwise doing something to make the name appear unique. This tag is optional

SetDefaultProperty specifies the name of the Default Property to receive the encoded value. You should ensure that this name does not match the name of other Substitution Values or properties from the Data Object to keep any name clashes from occurring. We recommend making the name all uppercase (case is sensitive) or otherwise doing something to make the name appear unique. This tag is optional

Specifying Multiple Actions

As noted above, you can chain multiple actions in the same XML Request by simply adding them together:

```
<Request>
 <SetSessionProperty>
    <SessionProperty name=’S_USERID’><Value>HEB</Value></SessionProperty>
    <SessionProperty name=’S_DEPTID’><Value>AA1A</Value></SessionProperty>
  </SetSessionProperty>
 <ExecuteSQL>
  UPDATE service_orders SET customer_quote = ‘${NEW_QUOTE}’ where order_number = ‘${order_number}’
 </ExecuteSQL>
 <ExecuteSQL>
  UPDATE service_order_counts SET num_sets = (num_sets+1) where order_number = ‘${order_number}’
 </ExecuteSQL>
 <ExecuteJDBCProgram>
  <Name>update_product_price</Name>
  <ParameterList>
   <Parameter type=’String’ length=’10’ direction=’IN’>
    <Value><![CDATA[${PRODUCT_NUMBER}]]></Value>
   </Parameter>
   <Parameter type=’Decimal’ length=’8’ scale=’2’ direction=’IN’>
    <Value><![CDATA[${NEW_PRICE}]]></Value>
   </Parameter>
  </ParameterList>
 </ExecuteJDBCProgram>
</Request>
```

The actions will be performed in the order they appear in the XML Request.

Returning Errors and Session/User Properties

If you are performing an ExecuteJDBCProgram, ExecuteHttpAction, or ExecuteAS400Program in your XML Request, you can return user messages as well as set Session and User Properties as an XML response. You can accomplish this by either having your stored procedure return a formatted XML response, or you can create an ad-hoc response in your XML request.

Formatting an XML Response in your Stored Procedure

You can return a formatted XML response directly from your Stored Procedure. In order to achieve this, an INOUT parameter should be added to the program call. This parameter should be of type String and should contain a returnMessages=’true’ tag in the XML Request. This tells Xalt | Mobility which parameter it should read after the program call to check for messages and properties:

```
<Request>
 <ExecuteJDBCProgram>
  <Name>update_product_price</Name>
  <ParameterList>
   <Parameter type=’String’ length=’10’ direction=’IN’>
    <Value><![CDATA[${PRODUCT_NUMBER}]]></Value>
   </Parameter>
   <Parameter type=’Decimal’ length=’8’ scale=’2’ direction=’IN’>
    <Value><![CDATA[${NEW_PRICE}]]></Value>
   </Parameter>
   <Parameter type=’String’ length=’2048’ scale=’0’ direction=’INOUT’ returnMessages=’true’>
    <Value><![CDATA[]]></Value>
   </Parameter>
  </ParameterList>
 </ExecuteJDBCProgram>
</Request>
```

An example of The XML Response string that should be returned by the program call is:

```
<Response>
 <Message type='error'>
  <Text><![CDATA[Pulse must be between 0 and 300.]]></Text>
  <DetailedText><![CDATA[This is more detailed text for the message ]]></DetailedText>
  <PropertyList>
   <Property name='ACCOUNT_NAME' />  <--- Look for param name first, then like property name next
   <Property name='P_NEW_AMOUNT' />
  </PropertyList>
 </Message>
 <Message type='error'>
  <Text><![CDATA[This is another error message]]></Text>
 </Message>
 <Message type='information'>
  <Text><![CDATA[This is an informational message]]></Text>
 </Message>
 <Message type='warning'>
  <Text><![CDATA[Systolic pressure of 370 is above 250.]]></Text>
  <DetailedText><![CDATA[This is more detailed text for the message ]]></DetailedText>
 </Message>
</Response>
```

Type should be ‘error’, ‘warning’, or ‘information’. Text should contain the message text. DetailedText (optional) can be used to specify second-level information for the message. PropertyList (optional) can be used to specify the Properties that contributed to the error or warning. The name of the property should match the Substitution Name of the Action Property. If a match is not found, it will attempt to match the name with a Like Property name of an Action Property.

If the program call processed successfully and there are no messages or properties to be returned, a null should be passed back in the parameter. The returnMessages parameter is only needed for programs that intend to pass back error and warning messages.

Xalt assumes that if error or warning messages are returned, then the program call must not have processed. Xalt will display the messages but leave the action prompt up so that the user can make changes to the parameters (if errors exist) and then resubmit the request. When the user resubmits the request, the Response XML that was originally sent to the client is sent back to the server in the returnMessages parameter (this is an INOUT parameter) so that the server can tell which warning messages were previously confirmed by the user.

Informational messages are handled slightly differently. If you return informational messages mixed with warnings and/or errors, the informational messages will be mixed in with the other messages on the action prompt. If you return only informational messages, the action will complete (closing the dialog) and a new dialog will appear containing the informational message(s). Before displaying, the dialog will inspect each informational message. If the message contains DetailedText, the DetailedText will be displayed to the user. If not, the Text will be displayed. Due to a current limitations, if informational messages are returned, the completion action will not be performed, as the informational message dialog will replace the completion action. Until this limitation is relaxed, you will want an action to EITHER return informational messages OR have a Completion Action, but not both.

Any extra XML passed from the program call to the client will be ignored and passed back. This allows the program call to pass extra information for tracking purposes. For example, the program call could pass back the following XML:

```
<Response>
  <Message type='warning'><Text>Diastolic pressure of 400 is above 250.</Text></Message>
  <Message type='warning'><Text>Are you sure you want to continue?</Text></Message>
  <PreviousValues>
    <Parm1><Value>720</Value></Parm1>
    <Parm2><Value>400</Value></Parm2>
    <Parm3><Value>370</Value></Parm3>
  </PreviousValues>
</Response>
```

If the user resubmits the action, this XML will be passed back to the program call. The program call could use the PreviousValues section to compare the current parameters passed in with the values passed in the previous time in order to further determine whether a warning message should be ignored or sent back to the user again.

Likewise, the program call can return an XML Response that can also set Session Properties and/or User Profile Properties as a result of the call using a similar syntax to setting these on the XML Request:

```
<Response>
  <SetSessionProperty>
      <SessionProperty name=’S_USERID’><Value>HEB</Value></SessionProperty>
      <SessionProperty name=’S_WHID’><Value>RTP</Value></SessionProperty>
   </SetSessionProperty>
  <SetUserProperty>
    <UserProperty name='U_USERID'><Value>HEB</Value></UserProperty>
   <UserProperty name='U_DEPTID'><Value>AA1A</Value></UserProperty>
   <RemoveUserProperty name='U_OLDUSERID'/>
  </SetUserProperty>
  <SetDefaultProperty>
   <DefaultProperty name='U_USERID'><Value>HEB</Value></DefaultProperty>
   <DefaultProperty name='U_DEPTID'><Value>AA1A</Value></DefaultProperty>
   <RemoveDefaultProperty name='U_OLDUSERID'/>
  </SetDefaultProperty>
</Response>
```

A common use for this is cases where a Stored Procedure call is performing a Create on an object with an auto-generated key. The Stored Procedure can pass the key values back as Session Properties, where the Completion Action can use them to redirect to the object after the create is completed.

Creating an ad-hoc XML Response

On a program call parameter that is an OUT or INOUT, you can also dynamically return messages and set session/ user properties based on the return value. This allows you to do things similar to the current returnMessages='true' setting, but on Stored Procedures and AS/400 program calls that you do not control or that were written prior to your involvement. Note that you cannot mix methods within a single program call. You can either use the formatted response method or the ad-hoc method, but not both on the same program call.

The syntax for taking a return value and turning it into a message is:

```
<Parameter type='String' length='255' direction='OUT'>
  <Value><![CDATA[]]></Value>
  <ReturnMessage type='Information'/>
</Parameter>
```

- The return value of the parameter will be the entire message.
- Type can be Error, Warning, or Information.
- For example, in this case the return value from the program is 'The current price is $3.50'. The connector gateway will take the value and format the correct XML around it to send back to Xalt.
- You can specify the ReturnMessage tag on multiple OUT or INOUT parameters. Each parameter will be formatted in a separate message.
- If the return value from the program is NULL or a blank string, a message will not be created.

The syntax for taking a return value and using it to formulate a message is:

```
<Parameter type='Decimal' length='8' scale='2' direction='INOUT'>
  <Value><![CDATA[${NEW_PRICE}]]></Value>
  <ReturnMessage type='Information'><![CDATA[The new value is: ${RETURN_VALUE}]]></ReturnMessage>
</Parameter>
```

- Adding a CDATA section to the ReturnMessage tag will allow you to create a formatted message
- Type can be Error, Warning, or Information.

- For example, in this case the return value from the program is '3.55'. The message returned is 'The new value is: 3.55'. The connector gateway will take the value and format the correct XML around it to send back to Xalt.
- ${RETURN_VALUE} is a special substitution value used to get the return value from the parameter and put it into the formatted message.
- For non-String fields (Decimals, Dates, etc), the standard toString() function will be applied to get the value for the message. If the return value is NULL, a blank string will be used.
- The message will be created unless the entire formatted value (message after substituting the return value) is blank.

On a program call parameter that is an OUT or INOUT, you can also dynamically set Session, User, and Default Properties based on the return value.

The syntax for taking a return value and using it to set a Session/User/Default property is:

```
<Parameter type='Decimal' length='8' scale='2' direction='INOUT'>
  <Value><![CDATA[${NEW_PRICE}]]></Value>
   <SetSessionProperty name='S_NEW_PRICE' />
   <SetUserProperty name='U_NEW_PRICE' />
   <SetDefaultProperty name='U_NEW_PRICE' />
</Parameter>
```

- For non-String fields (Decimals, Dates, etc), the standard toString() function will be applied to get the value for the property.
- Session/User properties will always be set. If the return value is NULL, a blank string will be used as the value.

The syntax for taking a return value and using it to formulate a Session/User/Default Property:

```
<Parameter type='Decimal' length='8' scale='2' direction='INOUT'>
  <Value><![CDATA[${NEW_PRICE}]]></Value>
   <SetSessionProperty name='S_NEW_PO'><![CDATA[PO${RETURN_VALUE}]]><SetSessionProperty/>
   <SetUserProperty name='U_NEW_PO'><![CDATA[PO #${RETURN_VALUE}]]><SetUserProperty/>
   <SetDefaultProperty name='U_NEW_PO'><![CDATA[PO #${RETURN_VALUE}]]><SetDefaultProperty/>
</Parameter>
```

- Adding a CDATA section to the SetSessionProperty, SetUserProperty, or SetDefaultProperty tag will allow you to format your own property value
- For example, in this case the return value is '3005'. The Session Property returned is 'PO3005' and the User Property returned is 'PO #3005'. The connector gateway will take the value and format the correct XML around it to send back to Xalt.
- ${RETURN_VALUE} is a special substitution value used to get the return value from the parameter and put it into the formatted property.
- For non-String fields (Decimals, Dates, etc), the standard toString() function will be applied to get the value for the message.
- Session/User/Default properties will always be set. If the return value is NULL, a blank string will be used as the value.

You can combine ReturnMessage, SetSessionProperty, SetUserProperty, and SetDefaultProperty tags for a Parameter, meaning that all 3 tags can be used on the same parameter. Each individual tag can only be used once, however (i.e. you cannot have 2 SetSessionProperty tags on the same parameter).

Accessing Return Values from Programs

Some databases allow Stored Procedures to define return types from functions instead of OUT parameters. We have added syntax to be able to access this value.

```
<ExecuteJDBCProgram>
    <Name>update_product_price</Name>
    <ParameterList>
        .....
    </ParameterList>
    <ReturnValue type='Integer'>
      <ReturnMessage type='Information'><![CDATA[The new value is: ${RETURN_VALUE}]]></ReturnMessage>
      <SetSessionProperty name='S_NEW_RETURN_VALUE'><![CDATA[PO-
${RETURN_VALUE}]]></SetSessionProperty>
      <SetUserProperty name='S_NEW_RETURN_VALUE'><![CDATA[PO-
${RETURN_VALUE}]]></SetUserProperty>
      <SetDefaultProperty name='S_NEW_RETURN_VALUE'><![CDATA[PO-
${RETURN_VALUE}]]></SetDefaultProperty>
    </ReturnValue>
  </ExecuteJDBCProgram>
```

- The <ReturnValue> tag indicates what will be done with the return value. The type attribute identifies the property type.
- As with Parameters, you can do ReturnMessage, SetSessionProperty, SetUserProperty, and SetDefaultProperty tags.
- You can instead do <ReturnValue type='Integer' **returnMessages='true'** /> to return formatted XML instead of doing ad-hoc values.

Return Values from HTTP Calls

You also have the ability to return messages and properties from HTTP Web Service Calls. We have added a <Return> tag to define this information:

```
<ExecuteHTTPAction>
  <URL>kabr.mybiz.net:8080</URL>
  <Method>POST</Method>
      ...
  <Return>
     ...
  </Return>
</ExecuteHTTPAction>
```

Since Web Service calls can return large formatted XML Responses, things are necessarily going to be more complicated that Stored Procedures. You will need to identify where to pull the messages/properties/values from using the XPath tag:

```
<XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CrtServItmLineResponse/ns:return]]></XPath>
```

Also, since most Web Services use Namespaces in their XML, and Namespaces require special handling to parse them out, you will need to include a Namespaces section identifying any namespaces if they exist in the XML Response:

```
<Namespaces>
  <Namespace name='soapenv'><![CDATA[http://schemas.xmlsoap.org/soap/envelope/]]></Namespace>
  <Namespace name='ns'><![CDATA[http://ws.apache.org/axis2]]></Namespace>
</Namespaces>
```

If your Web Service call returns a formatted JSON Response instead of an XML Response, you can replace XPath with JSONPath:

```
<JSONPath><![CDATA[/UserName]]></JSONPath>
```

The format of this expression is name/name/name… Array elements are expressed using [index]. A special value of [max] is available to get the last element. Wildcards and regular expressions are not currently supported.

```
Examples:  <JSONPath><![CDATA[/value/position/latitude]]></JSONPath>
           <JSONPath><![CDATA[/order/orderlines[0]/dueDate]]></JSONPath>
           <JSONPath><![CDATA[/3dmatrix[0][max][max]/value]]></JSONPath>
```

If your Web Service call returns a formatted XML Response, you can use the following syntax to retrieve the XML Response. Note the tag name is <ReturnMessages> to mimic the returnMessages='true' format of Stored Procedures:

```
  <ReturnMessages>
    <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:return]]>
    </XPath>
  </ReturnMessages>
```

As with Stored Procedures, you cannot mix <ReturnMessages> with <Message>, <SetSessionProperty>, <SetUserProperty>, and <SetDefaultProperty> tags.

Take return value and turn into an Informational message (can also choose Error and Warning):

```
  <Message type='Information'>
    <XPath>
      <![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:return]]>
    </XPath>
  </Message>
```

- This is similar to the <ReturnMessage> tag in Stored Procedures. You can specify a Type of Information, Warning, or Error.
- The return value of the parameter will be the message.
- For example, in this case the return value is 'The current price is $3.50'. The connector gateway will take that value and format the correct XML around it to send back to Xalt.
- You can have multiple Message tags. Each one will be formatted in a separate message and can return information from the XML Response.
- If the return value is NULL or a blank string, a message will not be created.

Take return value and use it to formulate an ad-hoc Return Message:

```
  <Message type='Information'>
    <XPath>
      <![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:return]]>
    </XPath>
    <Value><![CDATA[The new value is: ${RETURN_VALUE}]]></Value>
  </Message>
```

- This is similar to the <ReturnMessage> tag in Stored Procedures.
- For example, in this case the return value is '3.55'. The message returned is 'The new value is: 3.55'. The connector gateway will take the value and format the correct XML around it to send back to Xalt.
- ${RETURN_VALUE} is a special substitution value used to get the return value for the parameter.
- For non-String fields (Decimals, Dates, etc), the standard toString() function will be applied to get the value for the message. If the return value is NULL, a blank string will be used.
- The message will be created unless the entire formatted value (message after substituting the return value) is blank.
- If the return value is NULL or a blank string, a message will not be created.

Take return value and set it as a Session/User property:

```
<SetSessionProperty name='S_USERID'>
  <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CrtSrvItmLineResp/ns:userid]]></XPath>
</SetSessionProperty>
```

- For non-String fields (Decimals, Dates, etc), the standard toString() function will be applied to get the value for the message.
- Session/User properties will always be set. If the return value is NULL, a blank string will be used as the value.

Take return value and use it to formulate an ad-hoc Session/User/Default Property:

```
<SetSessionProperty name='S_DEPTID'>
 <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns: CrtSrvItmLineResp /ns:deptid]]></XPath>
 <Value><![CDATA[DEPT-${RETURN_VALUE}]]></Value>
</SetSessionProperty>
```

- Adding a Value tag to the SetSessionProperty, SetUserProperty, or SetDefaultProperty tag will allow you to format your own property value
- For example, in this case the return value is '3005'. The Session Property returned is 'DEPT-3005'. The connector gateway will take the value and format the correct XML around it to send back to Xalt.
- ${RETURN_VALUE} is a special substitution value used to get the return value from the parameter and put it into the formatted property.
- For non-String fields (Decimals, Dates, etc), the standard toString() function will be applied to get the value for the message.
- Session/User properties will always be set. If the return value is NULL, a blank string will be used as the value.

Putting it all together:

```
<ExecuteHTTPAction>
 <URL>http://kabr.mybiz.net:8080</URL>
 <Method>POST</Method>
 <ContentType>application/soap+xml; charset=utf-8</ContentType>
 <SetHeaderField name='Keep-Alive'>300</SetHeaderField>
 <SetHeaderField name='JSESSIONID'>${S_MYSESSIONID}</SetHeaderField>
 <Content><![CDATA[<?xml version="1.0" encoding="utf-8"?>
          <soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                  xmlns:xsd="http://www.w3.org/2001/XMLSchema"
                  xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
           <soap12:Body>
            <CreateContact xmlns=" http://kabr.mybiz.net:8080/">
             <inUser>${P_USER}</inUser>
            </CreateContact>
           </soap12:Body>
          </soap12:Envelope>]]></Content>
 <Return>
  <Namespaces>
   <Namespace name='soapenv'><![CDATA[http://schemas.xmlsoap.org/soap/envelope/]]></Namespace>
   <Namespace name='ns'><![CDATA[http://ws.apache.org/axis2]]></Namespace>
  </Namespaces>
  <ReturnMessages>   <!—Use ReturnMessage OR tags below, not both -->
   <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:return]]></XPath>
  </ReturnMessages>
  <Message type='Information'>
   <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:return]]></XPath>
  </Message>
  <Message type='Information'>
   <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:return]]></XPath>
   <Value><![CDATA[The new value is: ${RETURN_VALUE}]]></Value>
  </Message>
  <SetSessionProperty name='S_USERID'>
  <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:userid]]></XPath>
  </SetSessionProperty>
  <SetUserProperty name='U_DEPTID'>
  <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:deptid]]></XPath>
   <Value><![CDATA[DEPT-${RETURN_VALUE}]]></Value>
  </SetUserProperty>
  <SetDefaultProperty name='U_DEPTID'>
  <XPath><![CDATA[/soapenv:Envelope/soapenv:Body/ns:CreateServiceItemLineResponse/ns:deptid]]></XPath>
   <Value><![CDATA[DEPT-${RETURN_VALUE}]]></Value>
  </SetDefaultProperty>
 </Return>
</ExecuteHTTPAction>
```

Running Pre-Action Requests

There are times when you want to call a program before you bring up an Action’s prompt. For example, you may want to run a validation routine to ensure that the selected record is valid to perform the action against, or you may want to call a program to generate default values for your prompt. You have the ability to define a Pre-Action program in your Request XML. To recap, your Request XML currently has the following format:

```
<Request>Your action XML goes here</Request>
```

In order to use a Pre-Process request, you would change the Request XML to the following format:

```
<Action>
 <PreProcess>
  <Request> Your new pre-action request goes here </Request>
 </PreProcess>
 <Process>
  <Request>Your original post-action request goes here</Request>
 </Process>
</Action>
```

The PreProcess section will run before the prompt appears. The Process section will run at the end after the prompt is dismissed.

For example:

```
<Action>
 <PreProcess>
  <Request>
   <ExecuteJDBCProgram>
    <Name>usp_getnextid</Name>
     <ParameterList>
      <Parameter type='String' length='45' direction='IN'>
       <Value><![CDATA[${CURRENT_USER}]]></Value>
      </Parameter>
      <Parameter type='Integer' length='11' direction='OUT'>
       <Value><![CDATA[]]></Value>
       <SetSessionProperty name='S_NEXT_ID' />
     </Parameter>
    </ParameterList>
  </ExecuteJDBCProgram>
 </Request>
</PreProcess>
<Process>
 <Request>
 <ExecuteAS400Program>
  <Name>UPDPRDPRC</Name>
  <ParameterList>
   <Parameter type=’String’ length=’10’ direction=’IN’>
    <Value><![CDATA[${PRODUCT_NUMBER}]]></Value>
```

```
      </Parameter>
      <Parameter type='Decimal' length='8' scale='2' direction='IN'>
        <Value><![CDATA[${NEW_PRICE}]]></Value>
      </Parameter>
    </ParameterList>
  </ExecuteAS400Program>
 </Request>
</Process>
</Action>
```

Running Multiple Request Actions

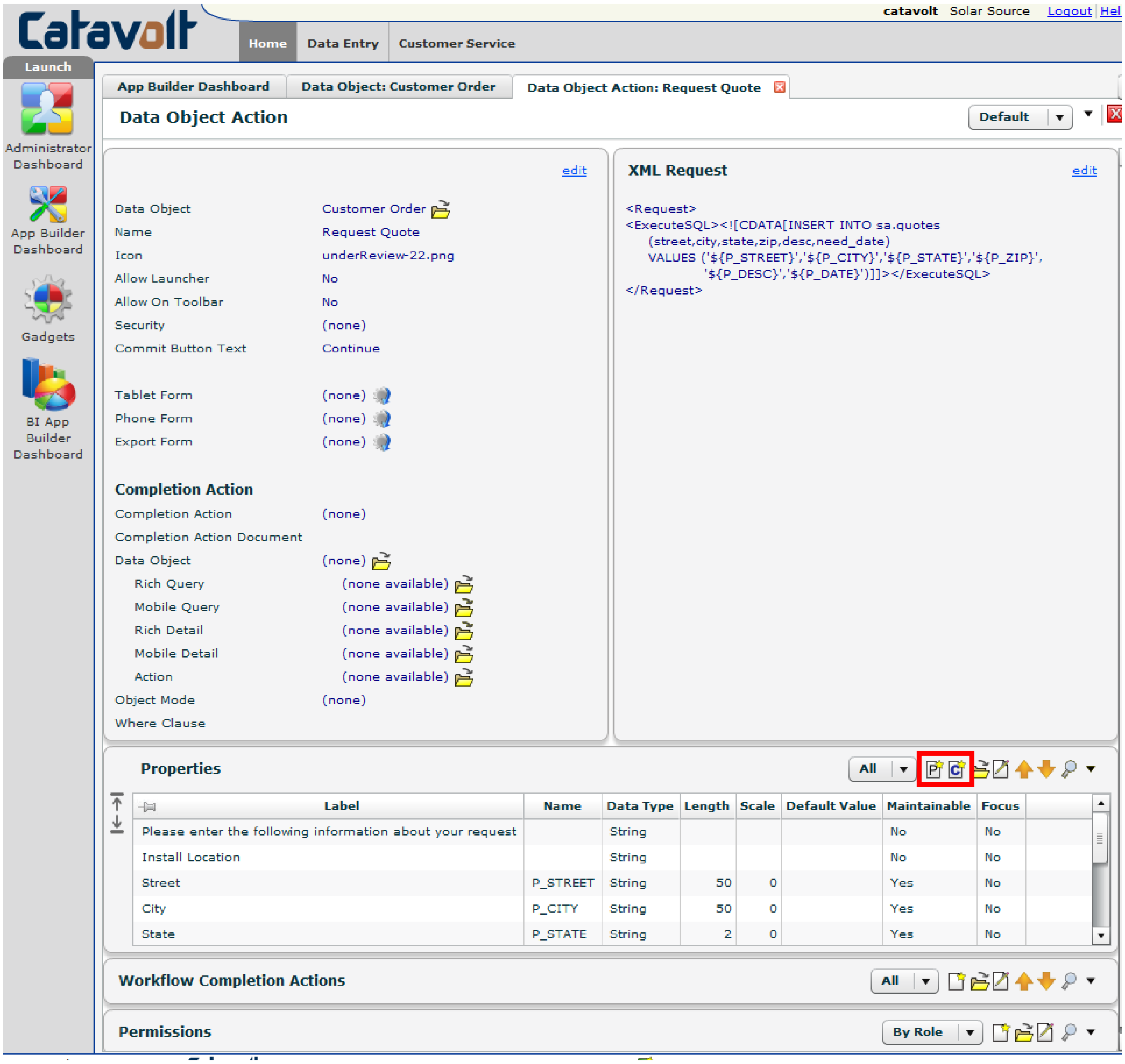
When performing the Request XML for an Action, all Substitution variables are replaced at once and the entire XML Request is sent to the Satellite Server for processing. This can be an issue if you need to perform multiple calls in a single Request and chain the results together (i.e. a Session Property set as the result of an <ExecuteJDBCProgram> call needs to be used in a later <ExecuteSQL> call within the same <Request>). In order to resolve this issue, you can specify multiple <Requests> within a Request XML if needed:

```
<Action>
  <PreProcess>
    <Request> Pre-Action 1 </Request>
    <Request> Pre-Action 2 </Request>
  </PreProcess>
  <Process>
   <Request> Action 1 </Request>
   <Request> Action 2 </Request>
  </Process>
</Action>
```

Each <Request> will have its Substitution values replaced, the request will be sent to the Satellite Server, performed, and any Session Properties returned before the next <Request> is processed. If your action supports multiple selections, all <Request> tags will be performed on a given selected record before the next record is processed. Using multiple <Request> tags allows you to use the results of one <Request> on a subsequent <Request> all within the same Action (at the cost of multiple round-trip calls between the Cloud and Satellite Servers).

Properties

The Properties section allows you to construct a prompt to be displayed when the end user chooses to perform an Action. The prompt can display confirmation information to the user, request values to be entered by the user, or both. You can construct the prompt using the Add Constant and Add Property actions. Properties will appear in the Action Prompt in the same order they appear below. To change the order, select a Property and press the Move Up and Move Down toolbar buttons. Note that if you leave the Properties section empty, the action will not display a prompt, but will run immediately when the action is chosen.



Data Object Action

Data Object	Customer Order
Name	Request Quote
Icon	underReview-22.png
Allow Launcher	No
Allow On Toolbar	No
Security	(none)
Commit Button Text	Continue
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)

Completion Action

Completion Action	(none)
Completion Action Document	
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Action	(none available)
Object Mode	(none)
Where Clause	

XML Request

```
<Request>
<ExecuteSQL><![CDATA[INSERT INTO sa.quotes
    (street,city,state,zip,desc,need_date)
    VALUES ('${P_STREET}','${P_CITY}','${P_STATE}','${P_ZIP}',
            '${P_DESC}','${P_DATE}')]]></ExecuteSQL>
</Request>
```

Properties

Label	Name	Data Type	Length	Scale	Default Value	Maintainable	Focus
Please enter the following information about your request		String				No	No
Install Location		String				No	No
Street	P_STREET	String	50	0		Yes	No
City	P_CITY	String	50	0		Yes	No
State	P_STATE	String	2	0		Yes	No

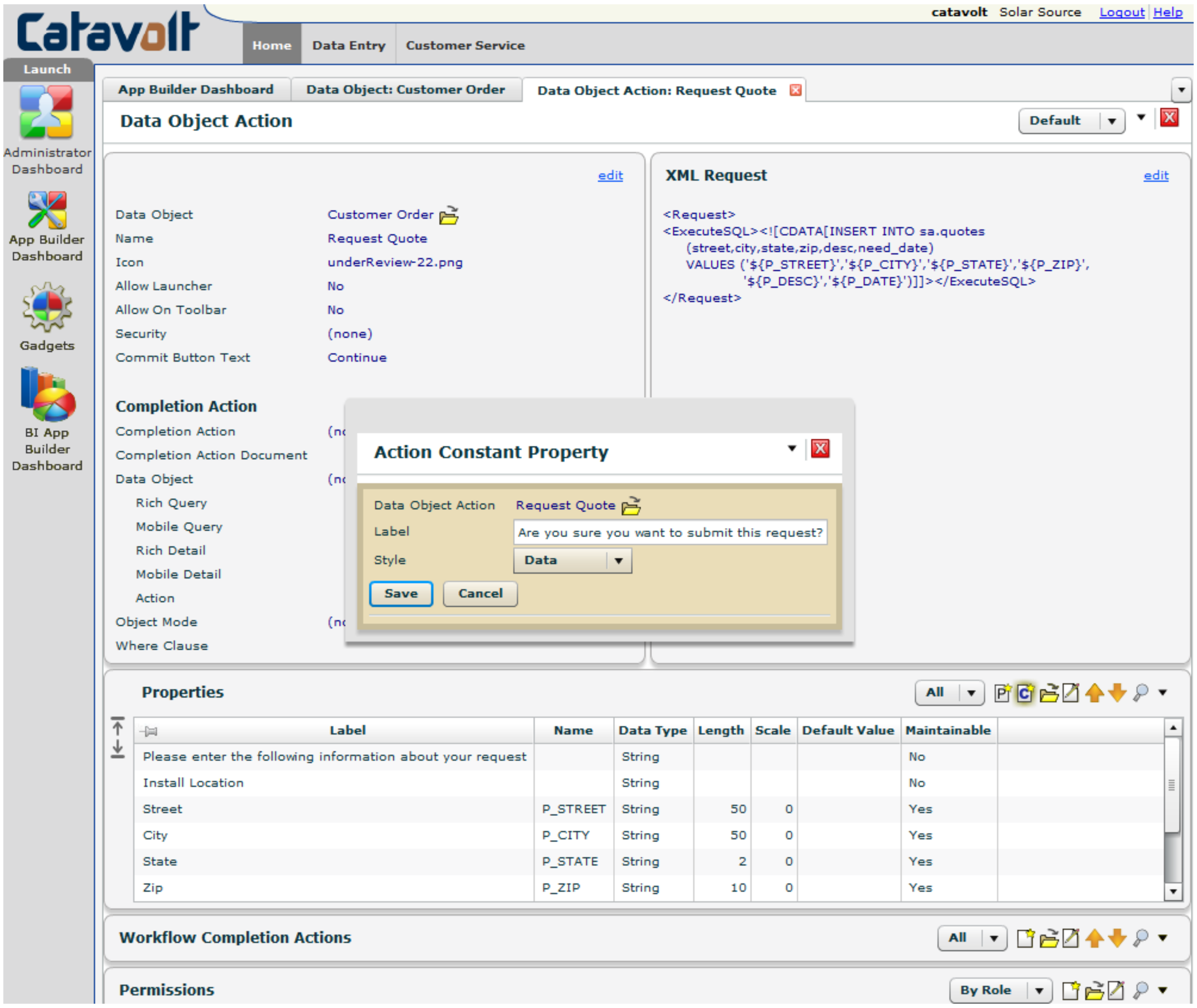
Workflow Completion Actions

Permissions

Figure 8: A Data Object Action details view with the Add Property and Add Constant actions highlighted

Adding Constants

Adding a constant to an Action Prompt is exactly the same as adding a constant to a Property Section:



Data Object Action

Data Object	Customer Order
Name	Request Quote
Icon	underReview-22.png
Allow Launcher	No
Allow On Toolbar	No
Security	(none)
Commit Button Text	Continue

Completion Action

Completion Action	(n
Completion Action Document	
Data Object	(n
Rich Query	
Mobile Query	
Rich Detail	
Mobile Detail	
Action	
Object Mode	(n
Where Clause	

XML Request

```
<Request>
<ExecuteSQL><![CDATA[INSERT INTO sa.quotes
    (street,city,state,zip,desc,need_date)
    VALUES ('${P_STREET}','${P_CITY}','${P_STATE}','${P_ZIP}',
              '${P_DESC}','${P_DATE}')]]></ExecuteSQL>
</Request>
```

Action Constant Property

Data Object Action	Request Quote
Label	Are you sure you want to submit this request?
Style	Data

Properties

Label	Name	Data Type	Length	Scale	Default Value	Maintainable
Please enter the following information about your request		String				No
Install Location		String				No
Street	P_STREET	String	50	0		Yes
City	P_CITY	String	50	0		Yes
State	P_STATE	String	2	0		Yes
Zip	P_ZIP	String	10	0		Yes

Workflow Completion Actions

Permissions

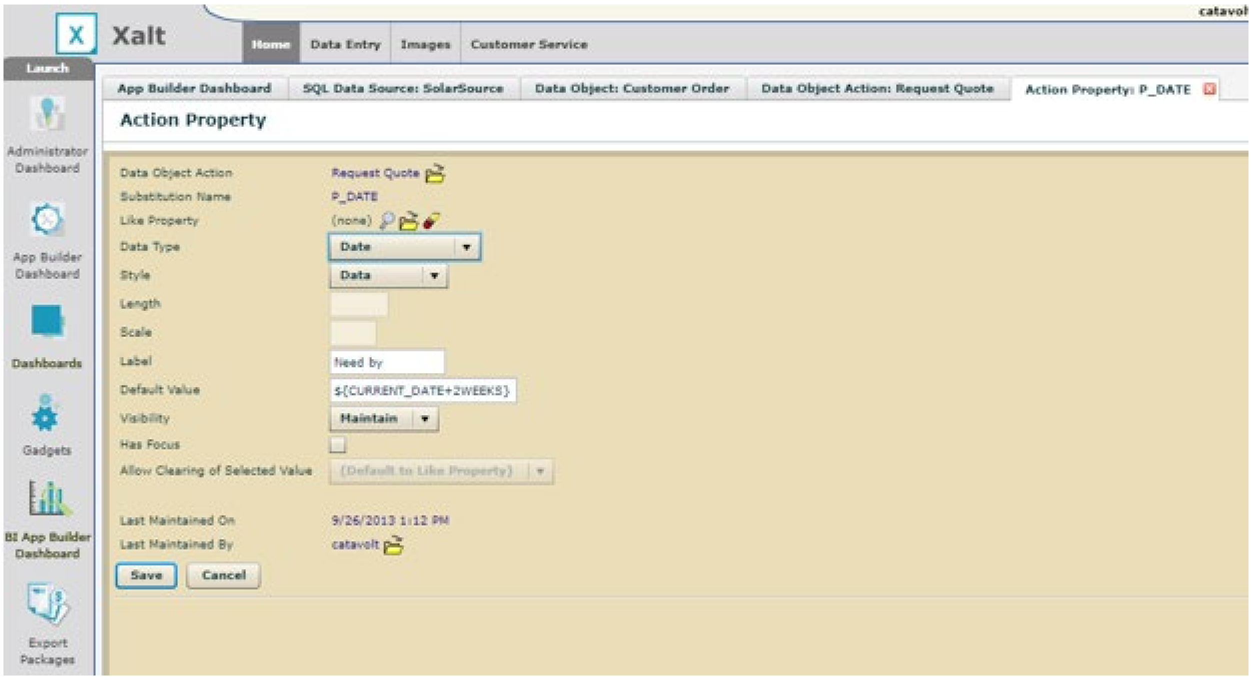
Figure 9: The Add Constant Property prompt

Label specifies the description that will be displayed on the prompt. You can leave this value empty in order to add a blank line to the prompt.

Style specifies the font that is used for displaying this value. Data presents it in a normal font, while Heading 1 through Heading 4 will display the value in bold with different font sizes (Heading 1 being the largest font size down to Heading 4).

Adding Properties

You can also add a property to the Action prompt. A property can be used to display data, request data, or both.



Action Property

Data Object Action	Request Quote
Substitution Name	P_DATE
Like Property	(none)
Data Type	Date
Style	Data
Length	
Scale	
Label	Need by
Default Value	${CURRENT_DATE+2WEEKS}
Visibility	Maintain
Has Focus	□
Allow Clearing of Selected Value	(Default to Like Property)
Last Maintained On	9/26/2013 1:12 PM
Last Maintained By	catavolt

Figure 10: The add Action Property view

When creating a Property, you must first specify a Substitution Name. This is the name you can use to reference the property when you are running the XML Request. For example:

```
<Request>
 <ExecuteSQL>
  UPDATE service_orders SET need_by_date = '${P_DATE}' where order_number = '${order_number}'
 </ExecuteSQL>
</Request>
```

You should ensure that this name does not match the name of other Substitution Values or properties from the Data Object to keep any name clashes from occurring. We recommend making the name all uppercase (case is sensitive) or otherwise doing something to make the name appear unique. If you specify a Like Property and leave this value blank, it will default to P_<Like Property Name>.

Like Property allows you to have this property adopt the formatting and behavior of an existing Defined Property for this Data Object. Some of the features that can be adopted are Property Values, Classes (URL, Password, etc), Length, Scale, etc.

Data Type specifies the parameter type. Allowable values are Boolean, Date, Decimal, Large Binary, Whole Number, String, Text, and Time. Note that if you have specified a Like Property, this value will default to the Data Type of the Like Property and be disabled.

Style specifies the font that is used for displaying this value. Data presents it in a normal font, while Heading 1 through Heading 4 will display the value in bold with different font sizes (Heading 1 being the largest font).

Length specifies the property's length. Note that if you have specified a Like Property, this value will default to the Length of the Like Property and be disabled.

Scale specifies the number of decimals for a numeric property. Note that if you have specified a Like Property, this value will default to the Scale of the Like Property and be disabled.

Label specifies the description that will be displayed for the property on the Action Prompt. Note that if you have specified a Like Property, this value will default to the Description of the Like Property.

Default Value specifies the value that should appear for this prompt when the Action Prompt is first displayed. You can leave this value blank to not provide a default value. Default Value can contain constants or substitution values. See Appendix A: Specifying Messages and Substitution Values for more information about how to create a Default Value. If you specify a Like Property, a typical usage of this field is to specify the value of the Like Property as the default value for the field.

Visibility specifies whether the value displayed in the prompt is to be hidden, read-only, or can be changed by the end user when the Action is performed. You may still refer to this property in the XML Request by its Substitution Name even if the value is Display or Hidden. By default, Visibility is set to Maintain.

Has Focus specifies whether the property displayed in the prompt should have input focus when the prompt is first displayed. Only one property can have Has Focus set to Yes. In addition, the property must have Maintainable set to Yes in order to use Has Focus. By default, Has Focus is set to No.

Allows Clearing of Selected Value specifies specifies that an empty (null) value should be added as an option to the property's dropdown list to allow the user to "unselect" a value for this property. There are 3 available options for this value:

- (Default to Like Property) – Use the "Allow Clearing of Selected Value" value on the Like Property to determine whether an empty value should be added.
- Yes – Add an empty value
- No – Do not add an empty value

The default value is (Default to Like Property). This field is disabled unless the property has a dropdown list (Like Property has at least one Property Value, the Like Property is the recipient of an Object Lookup with Combo Box, or the Like Property has a Native Data Type of Code Ref). Note that when using property substitution values (e.g. ${order_number}), Xalt will continue using the empty string as a substitution value instead of null.

Using Barcode Scanner / GPS Locator

Some mobile devices support the concepts of Barcode Scanning and/or GPS Location.   When these devices perform an Action, they look for the existence of special substitution variables such as ${CURRENT_BARCODE} or ${CURRENT_LOCATION.CITY}. If CURRENT_BARCODE is found within the action (as a default value, in the request XML, etc.), the mobile device will first open a Barcode Scanner (if available) and allow the user to scan a barcode. The resulting barcode string will be available in the CURRENT_BARCODE substitution value. Likewise, if one or more CURRENT_LOCATION substitution values are found, the mobile client will first open a GPS Locator panel (if available). The resulting GPS location of the mobile device will be available in the various CURRENT_LOCATION substitution values.

See Appendix A: Specifying Messages and Substitution Values for more information about using Action Substitution Values.

Using NFC Scanner

Some mobile devices support the concept of NFC Scanning. When these devices perform an Action, they look for the existence of a special substitution variable called ${CURRENT_NFC}. If CURRENT_NFC is found within the action (as a default value, in the request XML, etc.), the mobile device will first open an NFC Scanner (if available) and allow the user to scan a value from an NFC tag. The resulting string will be available in the CURRENT_NFC substitution value.

See Appendix A: Specifying Messages and Substitution Values for more information about using Action Substitution Values.

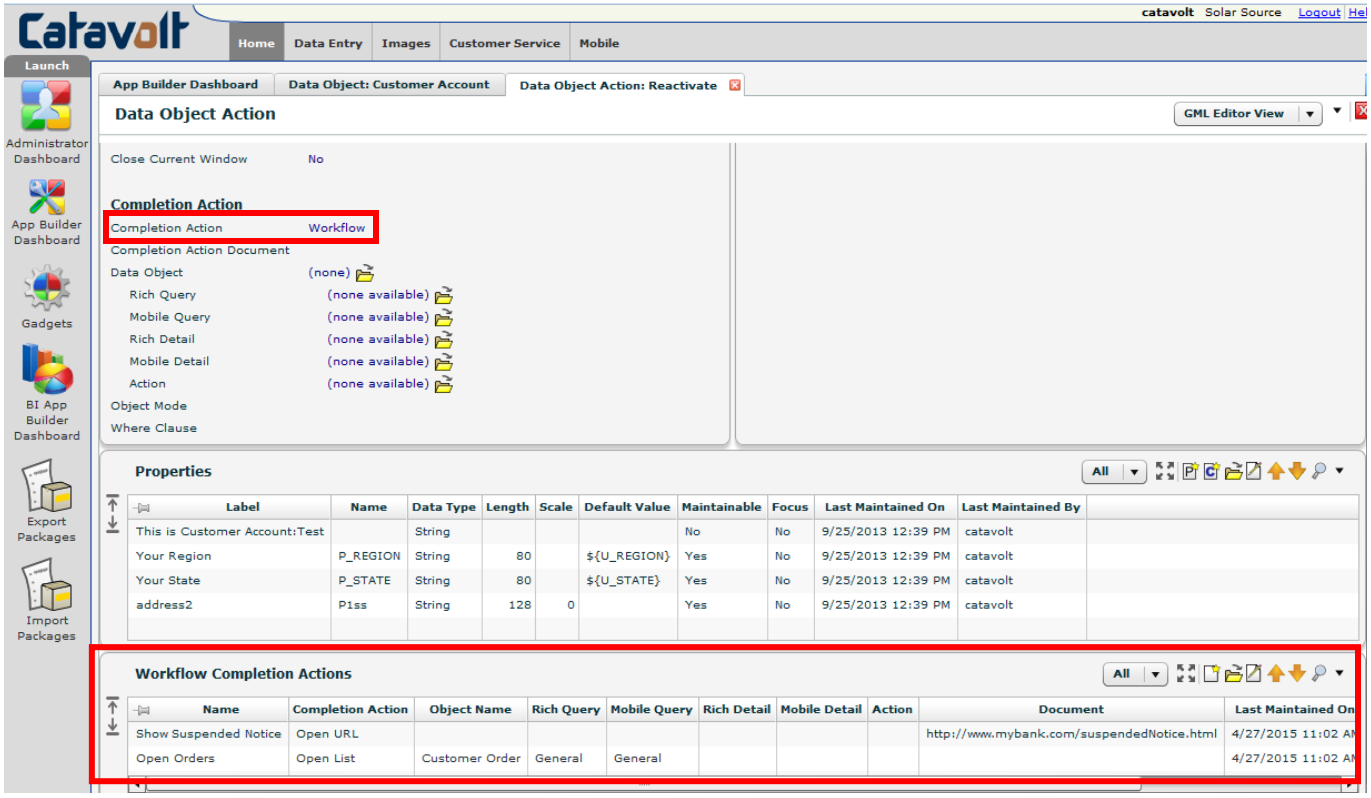
Using Map Drop Pins

When viewing a Map on a mobile device, the user has the ability to drop a pin on the Map by long pressing a location on the Map or by using the Address Search option. When performing an Action when a pin is dropped on a Map, special substitution variables are available to retrieve the location of the pin such as ${SELECTED_LOCATION.CITY}, ${SELECTED_LOCATION.LATITUDE}, etc. These substitution variables are available to use throughout the Action, such as in the Action Property, Request XML, Completion Action, etc.

See Appendix A: Specifying Messages and Substitution Values for more information about using Action Substitution Values.

Workflow Completion Actions

There are times where you may want an Action to run a different Completion Action based on the characteristics of the selected record or as a result of the action performed. You can accomplish this using Workflow Completion Actions. When you set the Completion Action property to Workflow, you have the ability to create multiple Workflow Completion Actions and tie them to rules that determine which Completion Action is performed when an Action completes. In cases where multiple Workflow Completion Actions pass the list of rules, the first Workflow Completion Action will be performed. To change the order, select a Workflow Completion Action and press the Move Up and Move Down toolbar buttons.



Data Object Action

Close Current Window No

Completion Action

Completion Action Workflow

Completion Action Document

Data Object (none)

Rich Query (none available)

Mobile Query (none available)

Rich Detail (none available)

Mobile Detail (none available)

Action (none available)

Object Mode

Where Clause

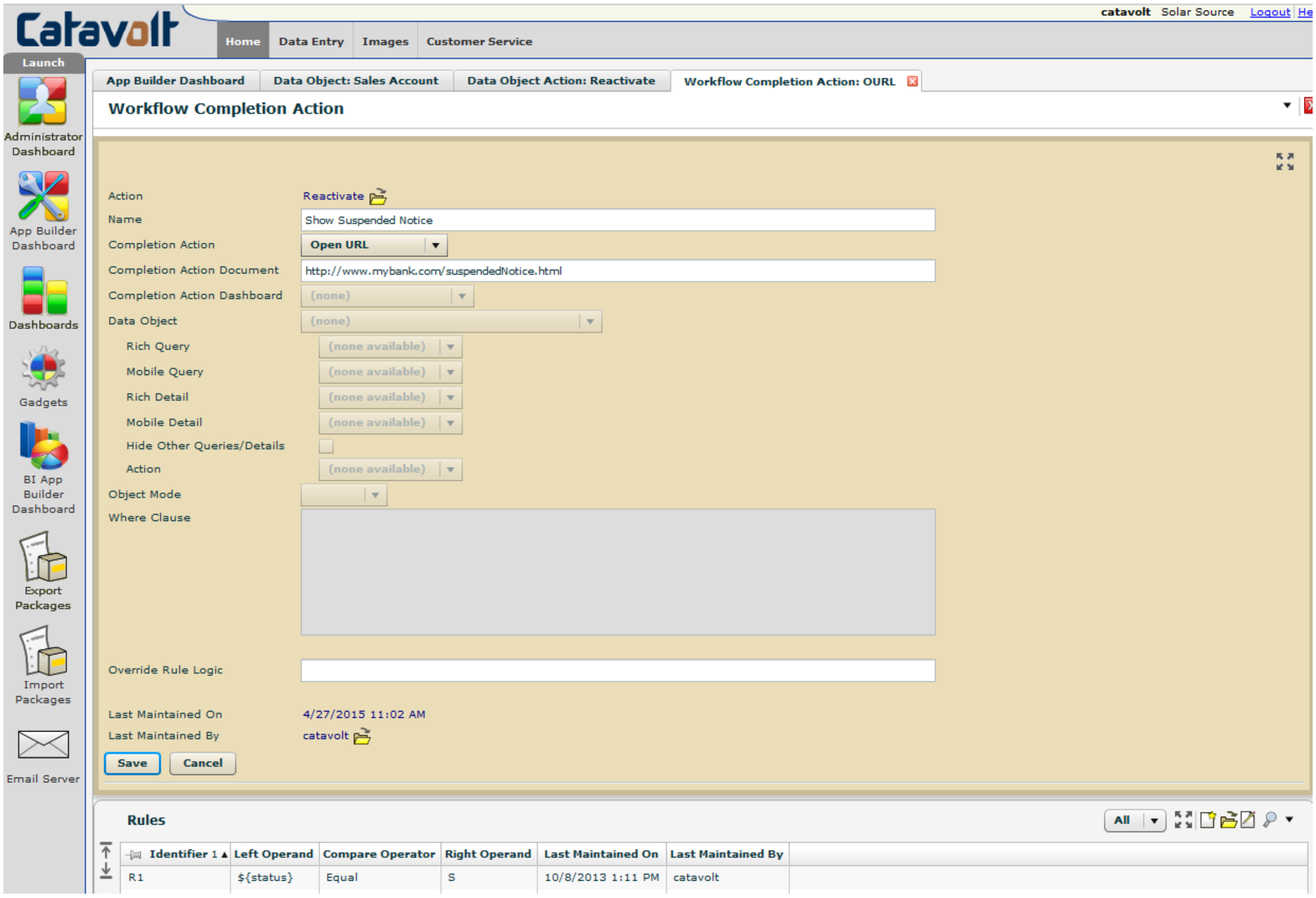
Properties

Label	Name	Data Type	Length	Scale	Default Value	Maintainable	Focus	Last Maintained On	Last Maintained By
This is Customer Account:Test		String				No	No	9/25/2013 12:39 PM	catavolt
Your Region	P_REGION	String	80		${U_REGION}	Yes	No	9/25/2013 12:39 PM	catavolt
Your State	P_STATE	String	80		${U_STATE}	Yes	No	9/25/2013 12:39 PM	catavolt
address2	P1ss	String	128	0		Yes	No	9/25/2013 12:39 PM	catavolt

Workflow Completion Actions

Name	Completion Action	Object Name	Rich Query	Mobile Query	Rich Detail	Mobile Detail	Action	Document	Last Maintained On
Show Suspended Notice	Open URL							http://www.mybank.com/suspendedNotice.html	4/27/2015 11:02 AI
Open Orders	Open List	Customer Order	General	General					4/27/2015 11:02 AI

Figure 11: The Data Object Action detail with Workflow Completion Action section highlighted



Action	Reactivate
Name	Show Suspended Notice
Completion Action	Open URL
Completion Action Document	http://www.mybank.com/suspendedNotice.html
Completion Action Dashboard	(none)
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Hide Other Queries/Details	□
Action	(none available)
Object Mode	
Where Clause	
Override Rule Logic	
Last Maintained On	4/27/2015 11:02 AM
Last Maintained By	catavolt

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand	Last Maintained On	Last Maintained By
R1	${status}	Equal	S	10/8/2013 1:11 PM	catavolt

Figure 12: The Workflow Completion Action create view

Name is used to uniquely identify the Workflow Completion Action and is not presented to the end user.

Completion Action allows you to navigate to some other location when the Action has completed. Currently, the following values are allowed:

- Open Dashboard – Open Dashboard will open a new tab and display a Dashboard.
- Open Document – Open Document will download a specified document via FTP and display it in the client (if Allow FTP is set to Yes for the Action's Data Source). You may also specify a private URL that will then be accessed via the Connector gateway and then downloaded and displayed in the client
- Open URL – Open URL will open a new browser tab and display the specified URL.
- Open List – Open List will open a new tab and display a Data Object Query.
- Open Object – Open Object will open a new tab and display a Data Object Detail in either read or update mode
- Open Action – Open Action will run the action you specify after the completion of the current action.
- Open EDE View – Open EDE View will open a Hexagon PPM EDE (Engineering Data Editor) View for display. This option is only available for Actions with an ODATA Data Source. Note: This option requires a <SetEDEConfiguration> section in your Request XML.
- Display Visualization Object Properties -- Display Visualization Object Properties allows you to display a list of properties for a selected object inside a Visualization Section. This option is only available if the Data Object contains a Detail Visualization Section. Note: This option requires a <SetVisualizationObjectPropertiesConfiguration> section in your Request XML.

Completion Action Document allows you to specify the path and name of the document that is to be downloaded/redirected to upon completion of the Action. If your Completion Action is Open URL, you should specify the full destination URL (e.g. http://www.google.com/images/srpr/logo3w.png). If your Completion Action is Open Document, the value of this field will be appended to the FTP Path field in the Data Source in order to get the full path to be downloaded via FTP from the back end system. If a URL is specified for Open Document, the target of the URL (an image, PDF, etc) will be downloaded from the Connector gateway and sent to the client. Completion Action Document can also contain substitution values. See Appendix A: Specifying Messages and Substitution Values for more information about how to create a Completion Action Document. This value is only allowed if the Completion Action is Open Document or Open URL.

Completion Action Dashboard allows you to specify the Dashboard you wish to display on completion of the Action. See Chapter 14: Dashboards for more information on creating Dashboards

Data Object specifies which Data Object you want to display either a list or object of or perform an action against. This value is only allowed if the Completion Action is Open Object, Open List, or Open Action.

Rich Query specifies which Query you want to display on the list when using a rich client. This value is only allowed if the Completion Action is Open List or (under certain circumstances) Open Action (see Allowing Multiple Selections above for more information).

Mobile Query specifies which Query you want to display on the list when using a mobile client. This value is only allowed if the Completion Action is Open List or (under certain circumstances) Open Action (see Allowing Multiple Selections above for more information).

Rich Detail specifies which Detail you want to display on the object when using a rich client. This value is only allowed if the Completion Action is Open Object.

Mobile Detail specifies which Detail you want to display on the object when using a mobile client. This value is only allowed if the Completion Action is Open Object.

Hide Other Queries/Details specifies whether the runtime Query/Detail dropdown list will contain just the selected Rich/Mobile Query/Detail or all non-hidden Queries/Details. If Hide Other Queries/Details is selected, only the selected Rich/Mobile Query/Detail will be available to choose in the Query/Detail dropdown list.

Action specifies which Action you want to perform on the selected data object. This value is only allowed if the Completion Action is Open Action.

Object Mode specifies to display the Detail in read-only or update mode when displaying an object. This value is only allowed if the Completion Action is Open Object.

Where Clause specifies extra criteria to use when querying an object. If the Completion Action is Open Object, then Where Clause must be written to return a single record when it is performed (same as when creating a Dashboard Launcher for a Data Object). If the Completion Action is Open List, Where Clause can be optionally specified to further restrict the list of records returned. This value is only allowed if the Completion Action is Open Object or Open

Override Rule Logic specifies more advanced logic to evaluate the Workflow Completion Action Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Workflow Completion Action to run. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

(R1 AND R2) OR (R3 AND R4)

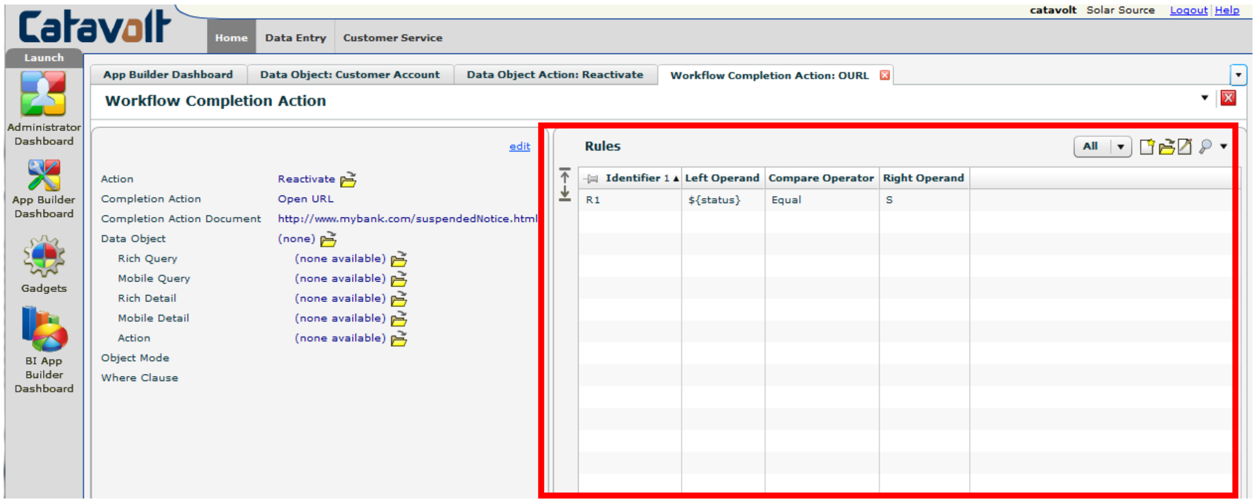
NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

Workflow Completion Action Rules

The Rules list on a Workflow Completion Action can be used to control when the Workflow Completion Action should be performed. A Workflow Completion Action can have zero or more rules. All rules must pass for the Workflow Completion Action to be performed. If a Workflow Completion Action has no rules, then it is considered to pass automatically.



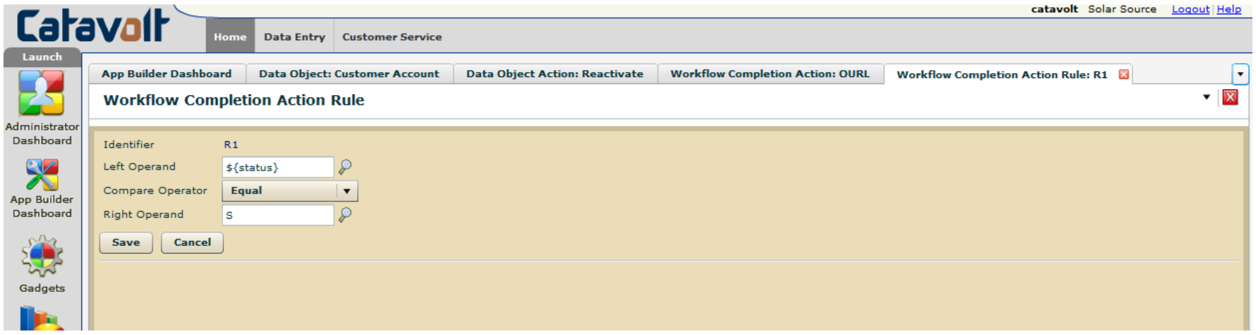
Workflow Completion Action

Action	Reactivate
Completion Action	Open URL
Completion Action Document	http://www.mybank.com/suspendedNotice.html
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Action	(none available)
Object Mode	
Where Clause	

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand
R1	${status}	Equal	S

Figure 13: Workflow Completion Action details view with the Rules query section highlighted


Catavolt
catavolt Solar Source Logout | Help
Home
Data Entry
Customer Service
Launch
Administrator Dashboard
App Builder Dashboard
Gadgets
App Builder Dashboard
Data Object: Customer Account
Data Object Action: Reactivate
Workflow Completion Action: OURL
Workflow Completion Action Rule: R1
Save
Cancel

Workflow Completion Action Rule

Identifier	R1
Left Operand	${status}
Compare Operator	Equal
Right Operand	S

Figure 14: The create Workflow Completion Action Rule view

When creating a Workflow Completion Action Rule, the Xalt will automatically create a unique Identifier for the Rule.

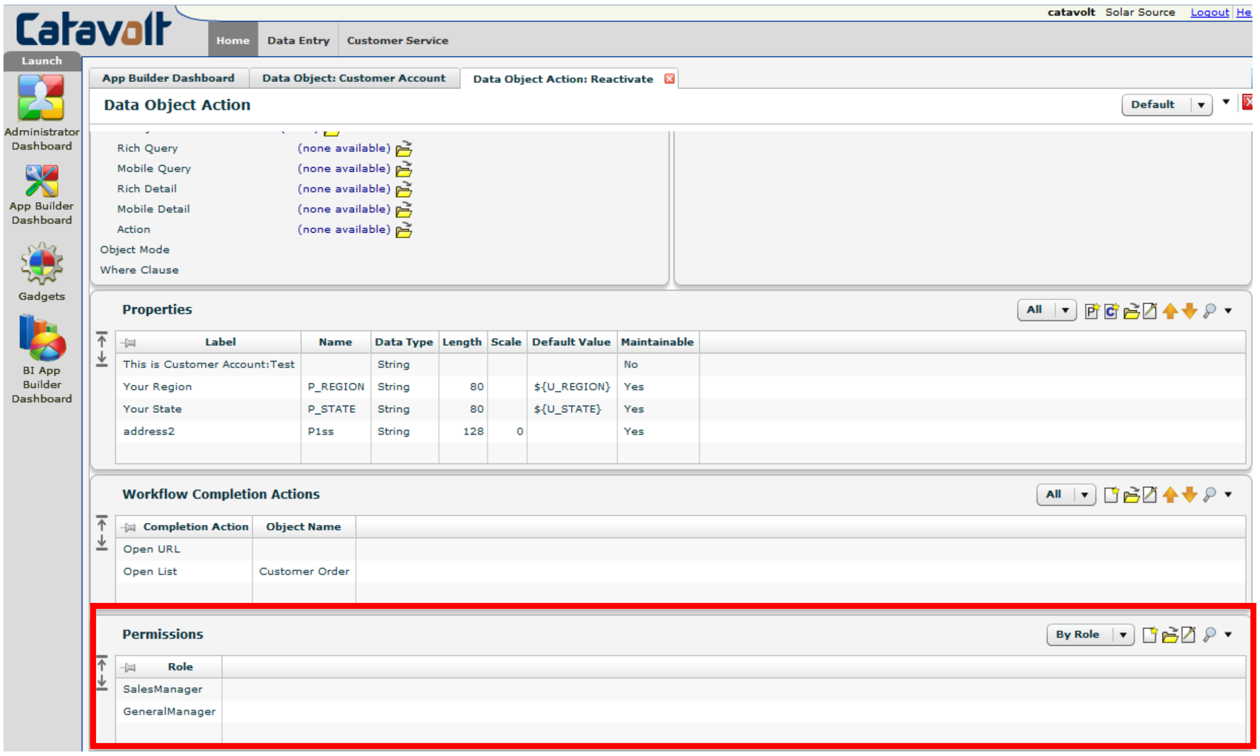
Left Operand and Right Operand specify the two values that will be compared with each other. You have 2 choices for operands: Constant Values (such as A or 1.00) or Substitution Values. You can enter a Substitution Value manually or press the Find button to have Xalt build a Substitution Value based on the Available Properties for the

Data Object. Appendix A: Specifying Messages and Substitution Values for more information about how to create Substitution Messages for Left Operand and Right Operand.

Compare Operator specifies how the two operands should be compared with each other. Options include Equal, Not Equal, Greater Than, Less Than, Greater or Equal, Less or Equal, Contains, Starts With, Ends With, Is Null, and Is Not Null.

Permissions

If you set Security on your Action to Use Authorization List, you can use the Permissions section to define which Security Roles have authority to perform the Action. The Permissions section will show a list of Security Roles that have been granted authority to your Action.



Data Object Action

Rich Query (none available)
Mobile Query (none available)
Rich Detail (none available)
Mobile Detail (none available)
Action (none available)
Object Mode
Where Clause

Properties

Label	Name	Data Type	Length	Scale	Default Value	Maintainable
This is Customer Account:Test		String				No
Your Region	P_REGION	String	80		${U_REGION}	Yes
Your State	P_STATE	String	80		${U_STATE}	Yes
address2	P1ss	String	128	0		Yes

Workflow Completion Actions

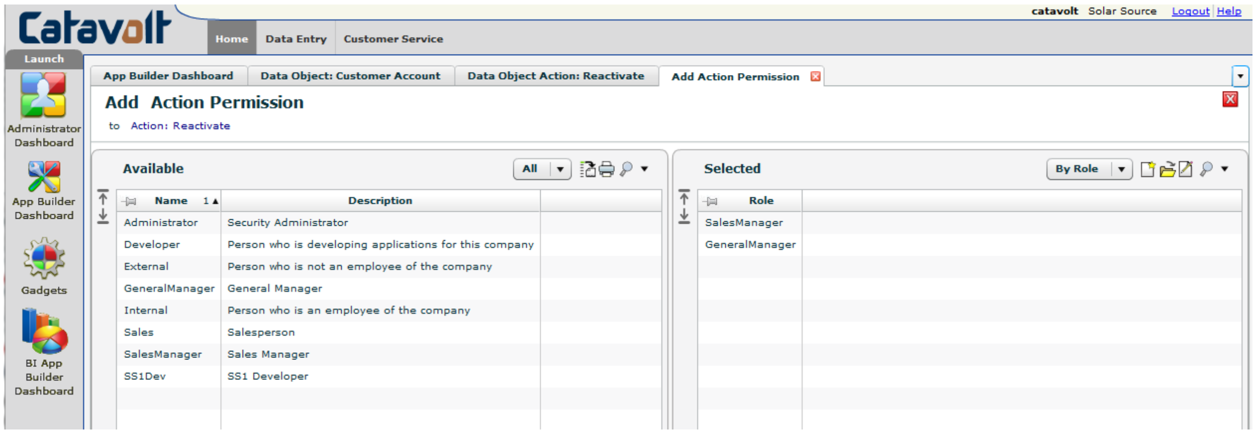
Completion Action	Object Name
Open URL	
Open List	Customer Order

Permissions

Role
SalesManager
GeneralManager

Figure 15: Data Object Action details with the Permissions section highlighted

When adding Data Object Action Permissions, you will be presented with two lists. The Available Permissions list shows all Security Roles. The Selected Permissions list shows the roles that have been granted authority to this Action.



Add Action Permission

to Action: Reactivate

Available

Name	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	SS1 Developer

Selected

Role
SalesManager
GeneralManager

Figure 16: The add Data Object Action Permission view.

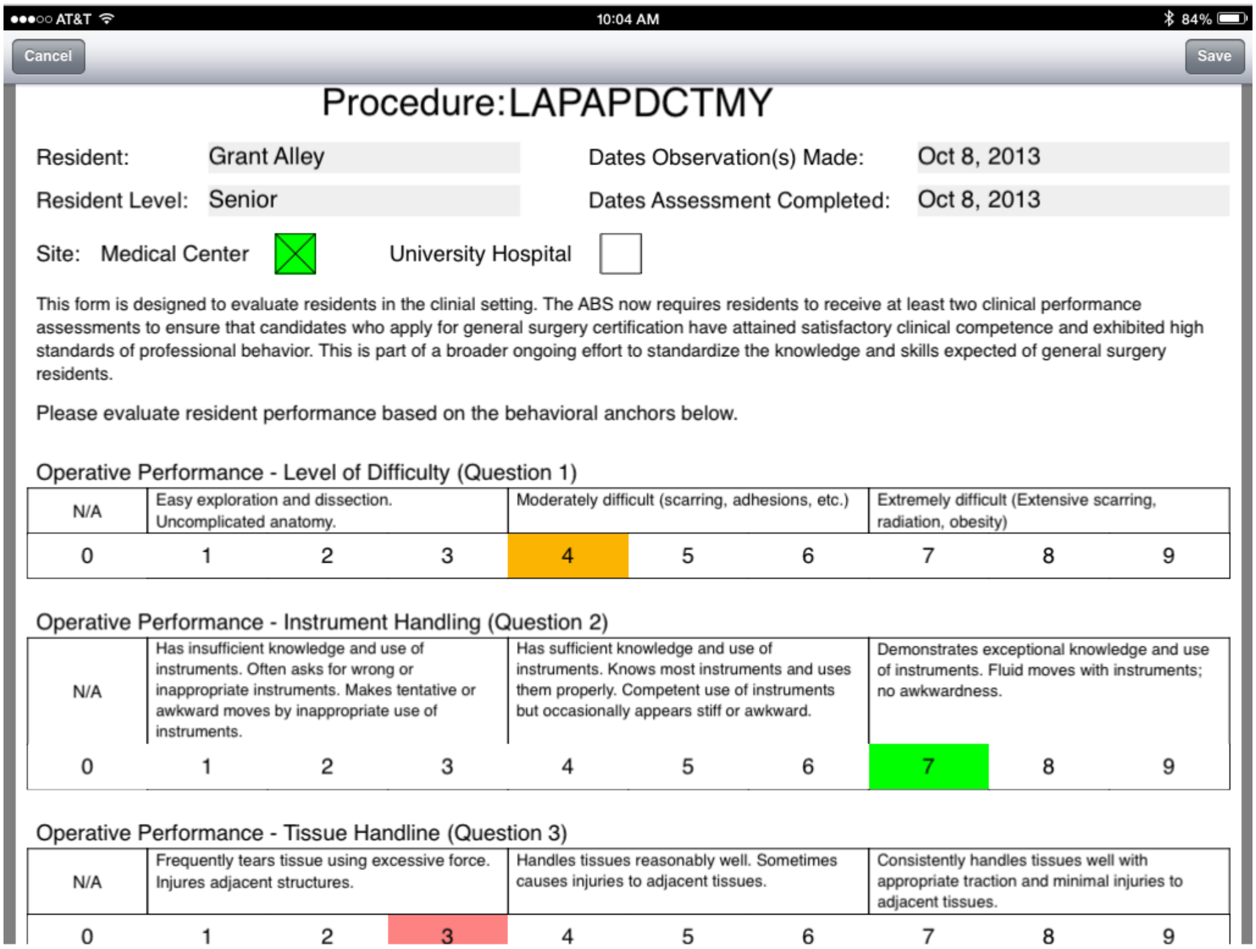
You can select a single or multiple Security Roles and press the Add button to add them to the Action.

See Chapter 10: Security for more information about object security in Xalt.

Other Data Object Action Menu Options

Upload Custom Form

Xalt | Mobility has the capability to render Action prompts as custom interactive forms. As you can see from the examples below, Action prompts can be displayed and edited using layouts and controls not available in the regular Xalt product.



Procedure:LAPAPDCTMY

Resident: Grant Alley

Dates Observation(s) Made: Oct 8, 2013

Resident Level: Senior

Dates Assessment Completed: Oct 8, 2013

Site: Medical Center ☒ University Hospital □

This form is designed to evaluate residents in the clinial setting. The ABS now requires residents to receive at least two clinical performance assessments to ensure that candidates who apply for general surgery certification have attained satisfactory clinical competence and exhibited high standards of professional behavior. This is part of a broader ongoing effort to standardize the knowledge and skills expected of general surgery residents.

Please evaluate resident performance based on the behavioral anchors below.

Operative Performance - Level of Difficulty (Question 1)

N/A	Easy exploration and dissection. Uncomplicated anatomy.			Moderately difficult (scarring, adhesions, etc.)			Extremely difficult (Extensive scarring, radiation, obesity)		
0	1	2	3	[✓ 4]	5	6	7	8	9

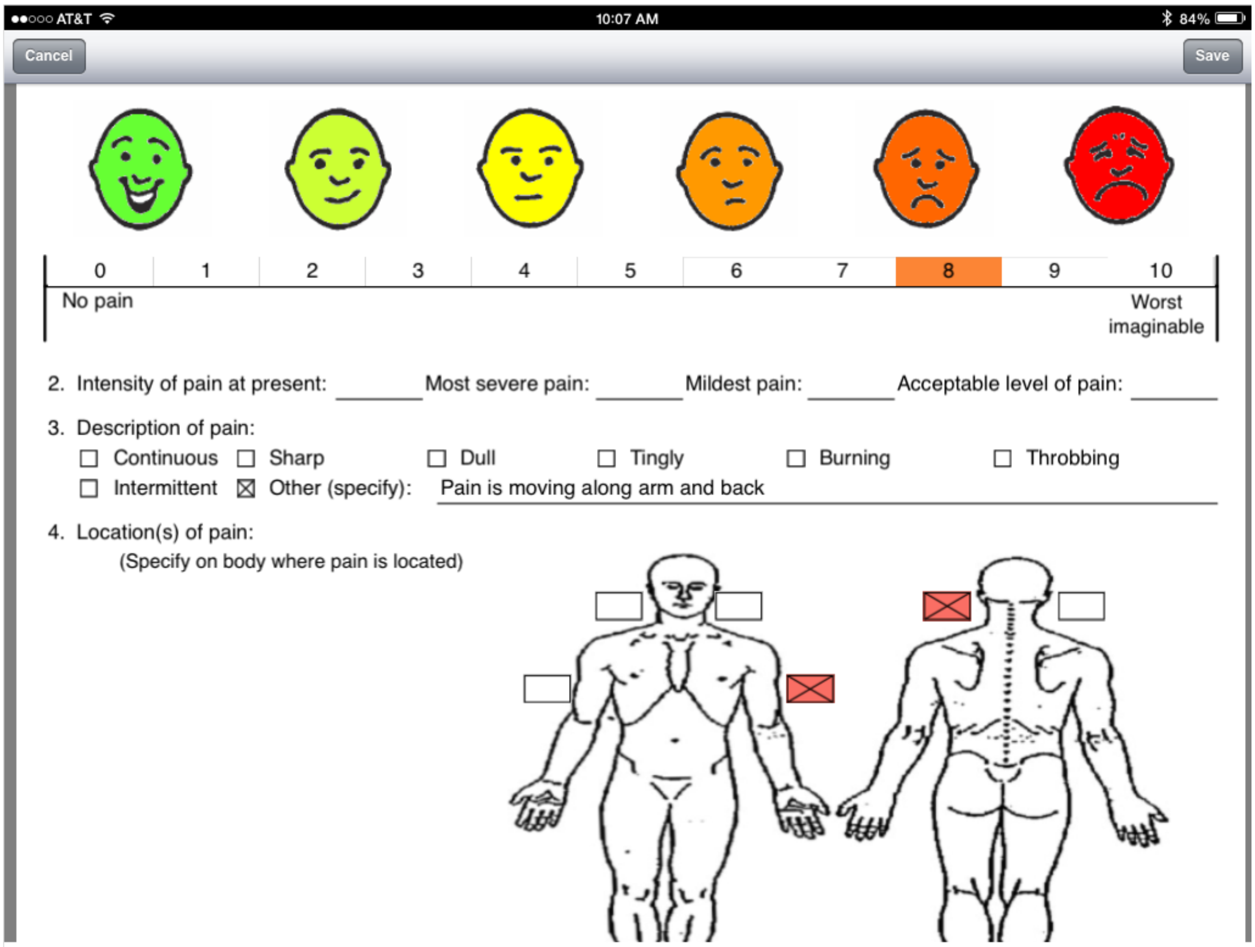
Operative Performance - Instrument Handling (Question 2)

N/A	Has insufficient knowledge and use of instruments. Often asks for wrong or inappropriate instruments. Makes tentative or awkward moves by inappropriate use of instruments.			Has sufficient knowledge and use of instruments. Knows most instruments and uses them properly. Competent use of instruments but occasionally appears stiff or awkward.			Demonstrates exceptional knowledge and use of instruments. Fluid moves with instruments; no awkwardness.		
0	1	2	3	4	5	6	[✓ 7]	8	9

Operative Performance - Tissue Handline (Question 3)

N/A	Frequently tears tissue using excessive force. Injures adjacent structures.			Handles tissues reasonably well. Sometimes causes injuries to adjacent tissues.			Consistently handles tissues well with appropriate traction and minimal injuries to adjacent tissues.		
0	1	2	[✓ 3]	4	5	6	7	8	9

Figure 17: An example of a Custom Form in Xalt



0	1	2	3	4	5	6	7	8	9	10
No pain										Worst imaginable

2. Intensity of pain at present: ______ Most severe pain: ______ Mildest pain: ______ Acceptable level of pain: ______

3. Description of pain:

☐ Continuous ☐ Sharp ☐ Dull ☐ Tingly ☐ Burning ☐ Throbbing

☐ Intermittent ☒ Other (specify): Pain is moving along arm and back

4. Location(s) of pain:

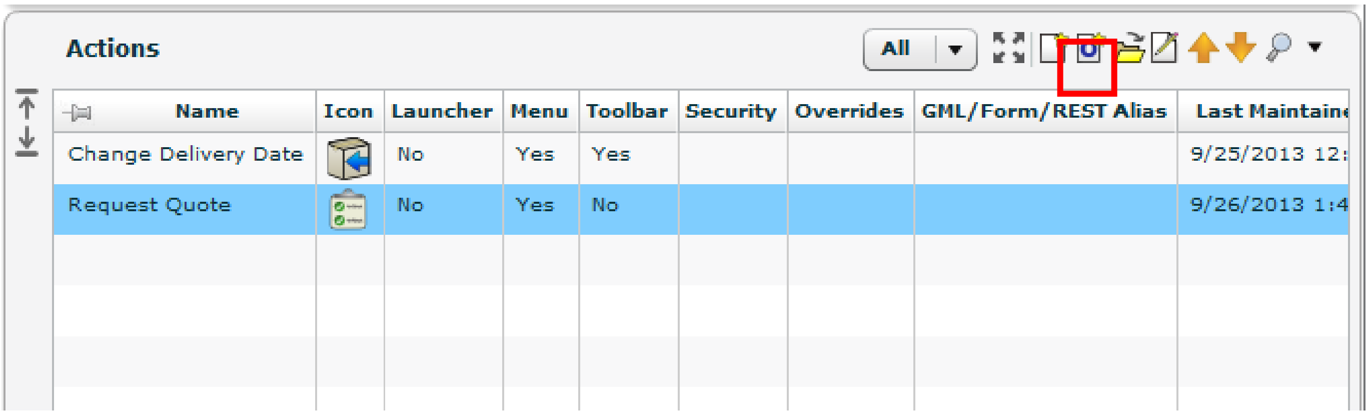
(Specify on body where pain is located)

Figure 18: An example of a Custom Form in Xalt

The Upload Custom Form menu option allows you to replace the existing Action prompt definition with an Xalt Custom Form. You can also view, upload, and remove custom forms from a dedicated launcher. See Chapter 13: Image/Asset Management for more information. Your Hexagon sales representative has more information on how to create and deploy Custom Forms.

Override Standard Action

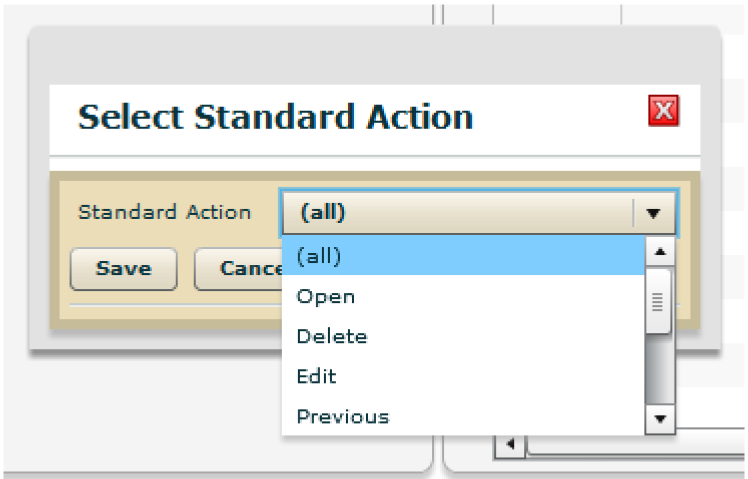
The Override Standard Action menu action allows you to override behavior on standard action such as Update, Delete, Refresh, etc.



Name	Icon	Launcher	Menu	Toolbar	Security	Overrides	GML/Form/REST Alias	Last Maintaine
Change Delivery Date		No	Yes	Yes				9/25/2013 12:
Request Quote		No	Yes	No				9/26/2013 1:4

Figure 19: The Actions list with the Override Standard Action button highlighted

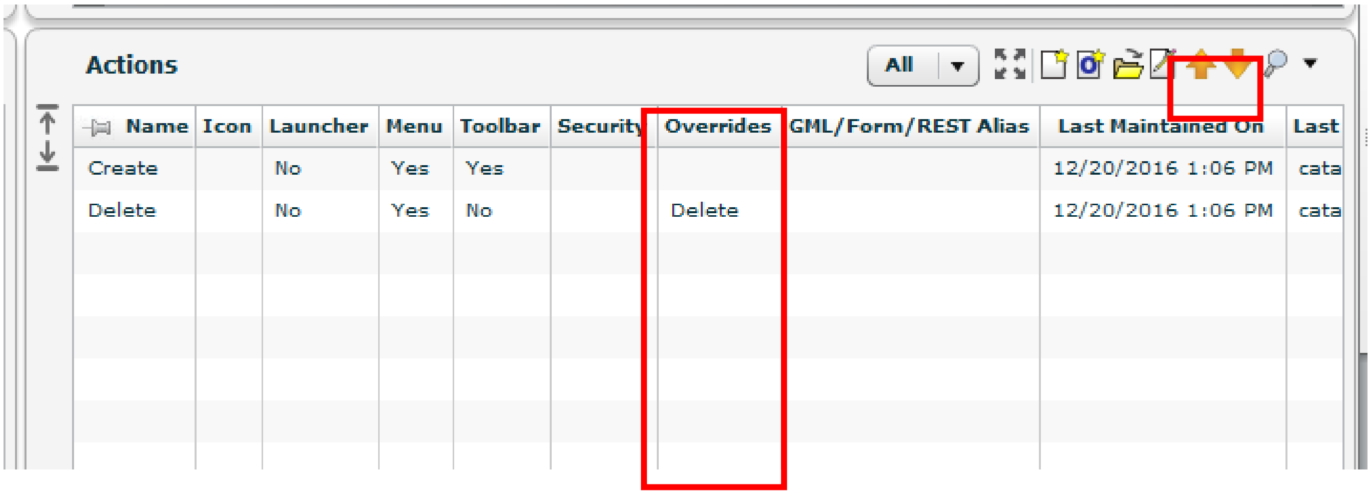
Choosing this action will bring up a dialog that contains a list of available standard actions that you can override. You can also choose (all) to add overrides for all standard actions. Adding an override will remove the action from the available list to ensure that you can only add 1 override per standard action. Note that standard actions are put in the available list if there is a possibility that they could be used by the Data Object. For example, Edit will not be available if the Data Object does not have Allow Update selected.


Select Standard Action

Standard Action: (all)

Figure 20: The Select Standard Action dialog

Adding an overridden standard action will put it at the end of the Actions list. The Overrides column has been added to identify the name of the standard action that has been overridden. You can arrange actions in whatever order you wish using the Move Up and Move Down buttons. You can mix and match ordering among standard and custom actions. Note that any standard actions that have not been overridden will continue to be placed in the menu/toolbar in their default locations.



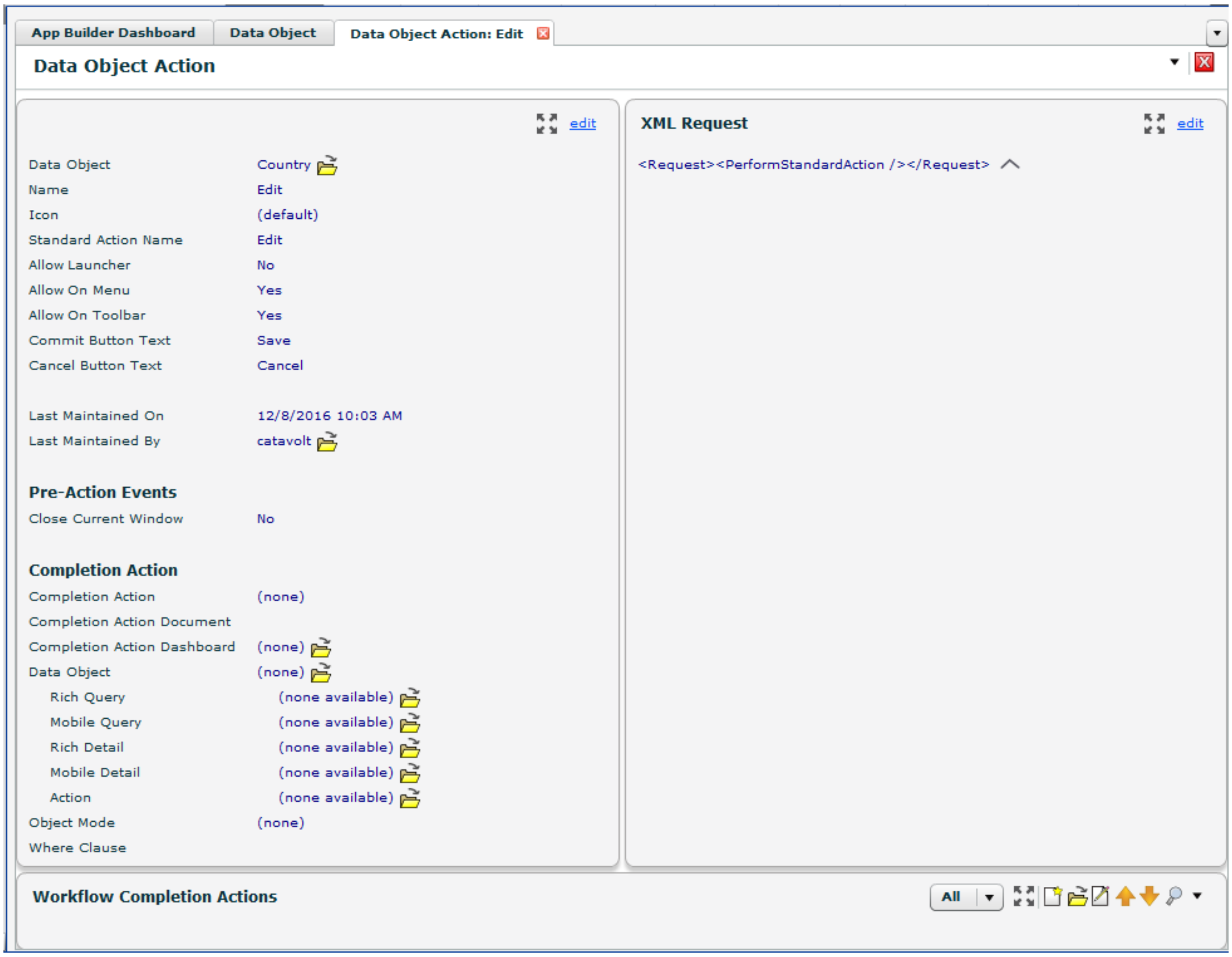
Name	Icon	Launcher	Menu	Toolbar	Security	Overrides	GML/Form/REST Alias	Last Maintained On	Last
Create		No	Yes	Yes				12/20/2016 1:06 PM	cata
Delete		No	Yes	No		Delete		12/20/2016 1:06 PM	cata

Figure 21: The Actions list with the Overrides column and Move Up / Move Down actions highlighted

Once you have overridden the standard action, you can then edit it like any other action, although there are some differences you will notice from custom actions. The view you see will depend on the standard action being overridden.

Create/Edit/Delete

The standard Create, Edit, and Delete actions will have a window like the following:



Data Object	Country
Name	Edit
Icon	(default)
Standard Action Name	Edit
Allow Launcher	No
Allow On Menu	Yes
Allow On Toolbar	Yes
Commit Button Text	Save
Cancel Button Text	Cancel
Last Maintained On	12/8/2016 10:03 AM
Last Maintained By	catavolt

Pre-Action Events

Close Current Window	No

Completion Action

Completion Action	(none)
Completion Action Document	
Completion Action Dashboard	(none)
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Action	(none available)
Object Mode	(none)
Where Clause	

```
<Request><PerformStandardAction /></Request>
```

Figure 22: The Action detail for the standard Create / Edit / Delete actions

You will be able to change the following:

Name – Change the menu and tip text for this action.

Icon – Change the icon for this action.

Allow Launcher – Create a Launcher for this action so that it can be called directly from a Workbench. This option is only available for the Create action. For other actions, this will be set to No and disabled.

Allow On Menu – Remove this action from the menu where desired. Note that you cannot use this to add the action on windows where it is not appropriate (for example, we do not put the Open action on the menu when you are already on a Details view).

Allow On Toolbar – Add or remove this action from the toolbar where desired. This is designed to remove unwanted actions from a toolbar, or to add an action on the menu to the toolbar if it does not already appear (for example, Delete).

Commit Button Text – Change the text on the Save button.

Cancel Button Text – Change the text on the Cancel button.

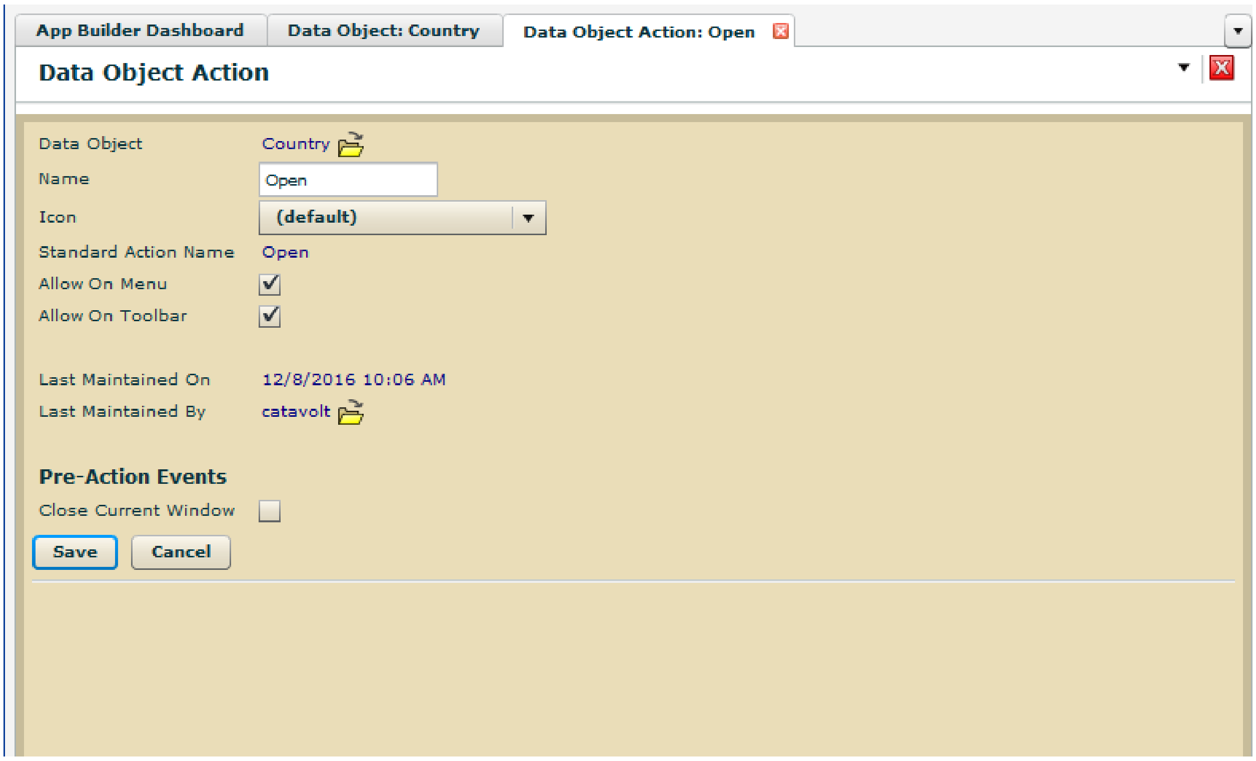
Close Current Window – Close the current window before running the action.

Completion Action – Define a Completion Action to occur when the Create/Update/Delete has completed successfully.

XML Request – Alter or enhance standard processing of the action (see **XML Request Options** below for more details).

Other Standard Actions

All other standard actions will have a much smaller array of options:



Data Object Action

Data Object	Country
Name	Open
Icon	(default)
Standard Action Name	Open
Allow On Menu	☑
Allow On Toolbar	☑
Last Maintained On	12/8/2016 10:06 AM
Last Maintained By	catavolt

Pre-Action Events

Close Current Window □

Figure 23: The Actions detail for other standard actions

Name – Change the menu and tip text for this action.

Icon – Change the icon for this action.

Allow On Menu – Remove this action from the menu where desired. Note that you cannot use this to add the action on windows where it is not appropriate (for example, we do not put the Open action on the menu when you are already on a Details view).

Allow On Toolbar – Add or remove this action from the toolbar where desired. This is designed to remove unwanted actions from a toolbar, or to add an action on the menu to the toolbar if it does not already appear (for example, Delete).

Close Current Window – Close the current window before running the action. Note that Close Current Window is only available for the Open action. For other standard actions it will be set to No and disabled.

XML Request Options

When you override a standard Create, Update, or Delete Action, the XML Request defaults to:

```
<Request><PerformStandardAction /></Request>
```

This will tell the action to perform its standard processing as if it were not overridden. As XML Requests for Standard Actions support many of the same capabilities as XML Requests for Custom Actions, you can alter this processing in a number of different ways. Let's use the Standard Edit action as an example.

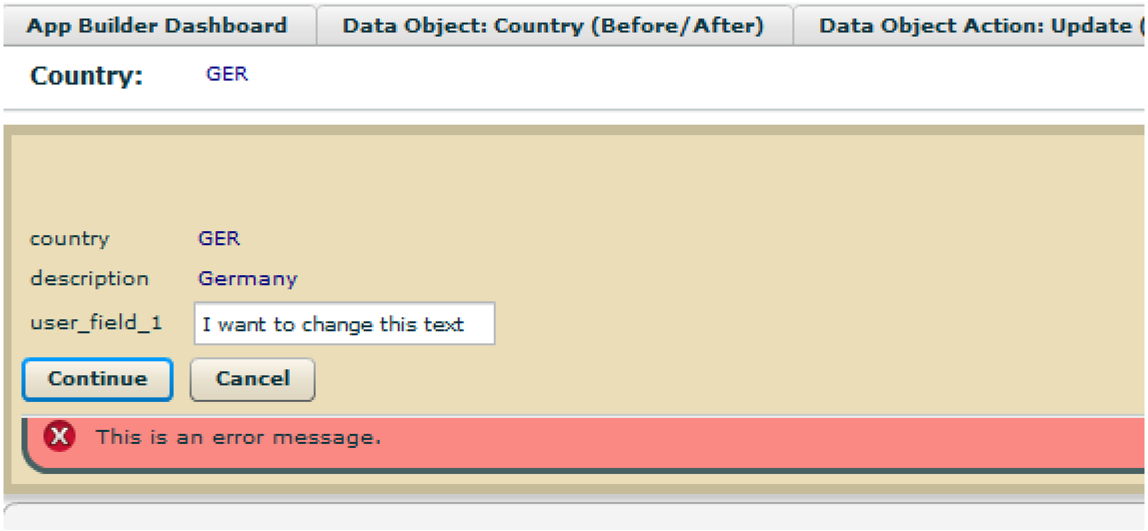
If you wish to perform some processing after the standard update occurs, put your XML in between the <PerformStandardAction /> and </Request> tags. In this case, we are going to insert a record into a tracking table after the standard update is performed:

```
<Request>
<PerformStandardAction />
<ExecuteSQL>INSERT INTO sa.log_table (user_id, description, date_written, time_written, timestamp_written)
VALUES ('${CURRENT_EFFECTIVE_USER}','${CURRENT_EFFECTIVE_USER} just updated Country ${country}.',
'${CURRENT_DATE@DATE}','${CURRENT_TIME@TIME}','${CURRENT_TIMESTAMP}')</ExecuteSQL>
</Request>
```

If you wish to perform some processing before the standard update occurs, put your XML in between the <Request> and <PerformStandardAction /> tags. In this case, we are going to call a validation routine before we call the standard update:

```
<Request>
<ExecuteJDBCProgram>
   <Name>sp_message_test</Name>
    <ParameterList>
      <Parameter type='Integer' length='10' direction='IN'><Value><![CDATA[1]]></Value></Parameter>
      <Parameter type='String' length='2048' direction='INOUT' returnMessages='true'>
        <Value><![CDATA[]]></Value>
      </Parameter>
   </ParameterList>
 </ExecuteJDBCProgram>
<PerformStandardAction />
</Request>
```

Note that if the pre-action fails because of an error, the standard update will not occur:


App Builder Dashboard
Data Object: Country (Before/After)
Data Object Action: Update (
Continue
Cancel

Country: GER

country GER
description Germany
user_field_1 I want to change this text

This is an error message.

Figure 24: A sample error message being returned from a pre-update validation call

If you wish to completely replace the standard processing with your own logic, simply replace the <PerformStandardAction /> tag with your own XML. In this case, we are going to call a stored procedure to perform the update instead of using standard processing:

```
<Request>
<ExecuteJDBCProgram>
  <Name>sp_update_country</Name>
  <ParameterList>
   <Parameter type='String' length='255' direction='IN'>
    <Value><![CDATA[${country}]]></Value>
   </Parameter>
   <Parameter type='String' length='255' direction='IN'>
    <Value><![CDATA[${description}]]></Value>
   </Parameter>
   <Parameter type='String' length='255' direction='IN'>
    <Value><![CDATA[${user_field_1}]]></Value>
   </Parameter>
   <Parameter type='String' length='255' direction='IN'>
    <Value><![CDATA[${user_field_2}]]></Value>
   </Parameter>
  </ParameterList>
</ExecuteJDBCProgram>
</Request>
```

If you wish to perform a pre-processing check (for example, validate that a record can be changed as soon as the user chooses the Edit action), use the existing <PreProcess> / <Process> tags for Actions. In this case, we are going to call a stored procedure that validates whether the selected Country can be changed. Note that this PreProcess call will be made as soon as the Edit action is taken (before the Update window comes up):

```
<Action>
<PreProcess>
<Request>
<ExecuteJDBCProgram>
  <Name>sp_pre_validate_country</Name>
  <ParameterList>
   <Parameter type='String' length='255' direction='IN'>
    <Value><![CDATA[${country}]]></Value>
   </Parameter>
   <Parameter type='String' length='4096' direction='INOUT' returnMessages='true'>
    <Value><![CDATA[]]></Value>
   </Parameter>
  </ParameterList>
</ExecuteJDBCProgram>
</Request>
</PreProcess>

<Process>
<Request>
<PerformStandardAction />
</Request>
</Process>
</Action>
```

If the validation stored procedure fails, the user will not be allowed to edit the record.

You are also free to mix and match, for example, running pre-validate together with a post-action after the update. Note that you can specify multiple <Request> tags if you need the results of one call to be used in a subsequent call (see Running Multiple Request Actions for more information of what multi-request Actions do).

If you use multiple <Request> tags, you must have your <PerformStandardAction /> tag outside of the <Request> tags, for example:

```
<Action>
 <Process>
   <PerformStandardAction />
   <Request>Request 1</Request>
   <Request>Request 2</Request>
 </Process>
</Action>
```

Chapter 7: Workbenches

Chapter Summary

A Workbench is a group of Launchers that are displayed together. Each Launcher displays either a list or detail view of an object, or can perform an action. Multiple Workbenches can be assigned to a user. When the user runs Xalt via a web browser, the list of Workbenches they have access to will appear across the top of the window. When a Workbench is selected, the list of Launchers for that Workbench will appear down the left side of the window. A workbench is also used to control the available launchers a user sees when they are using a mobile device.

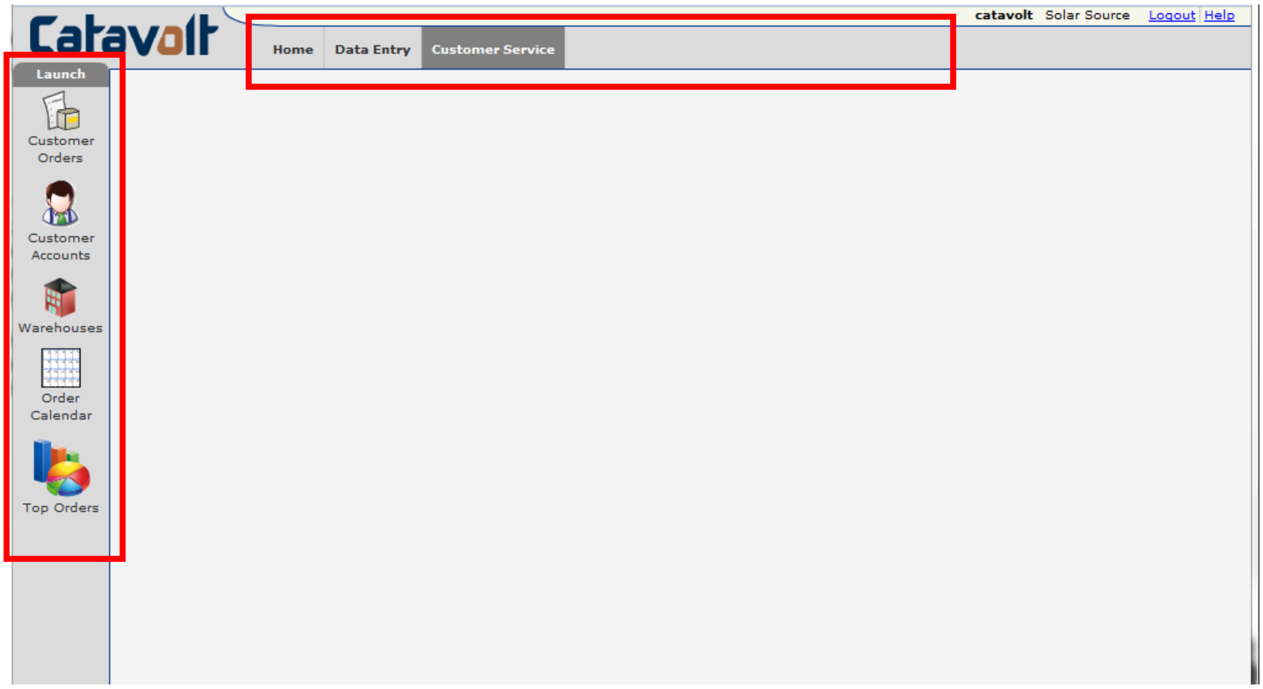
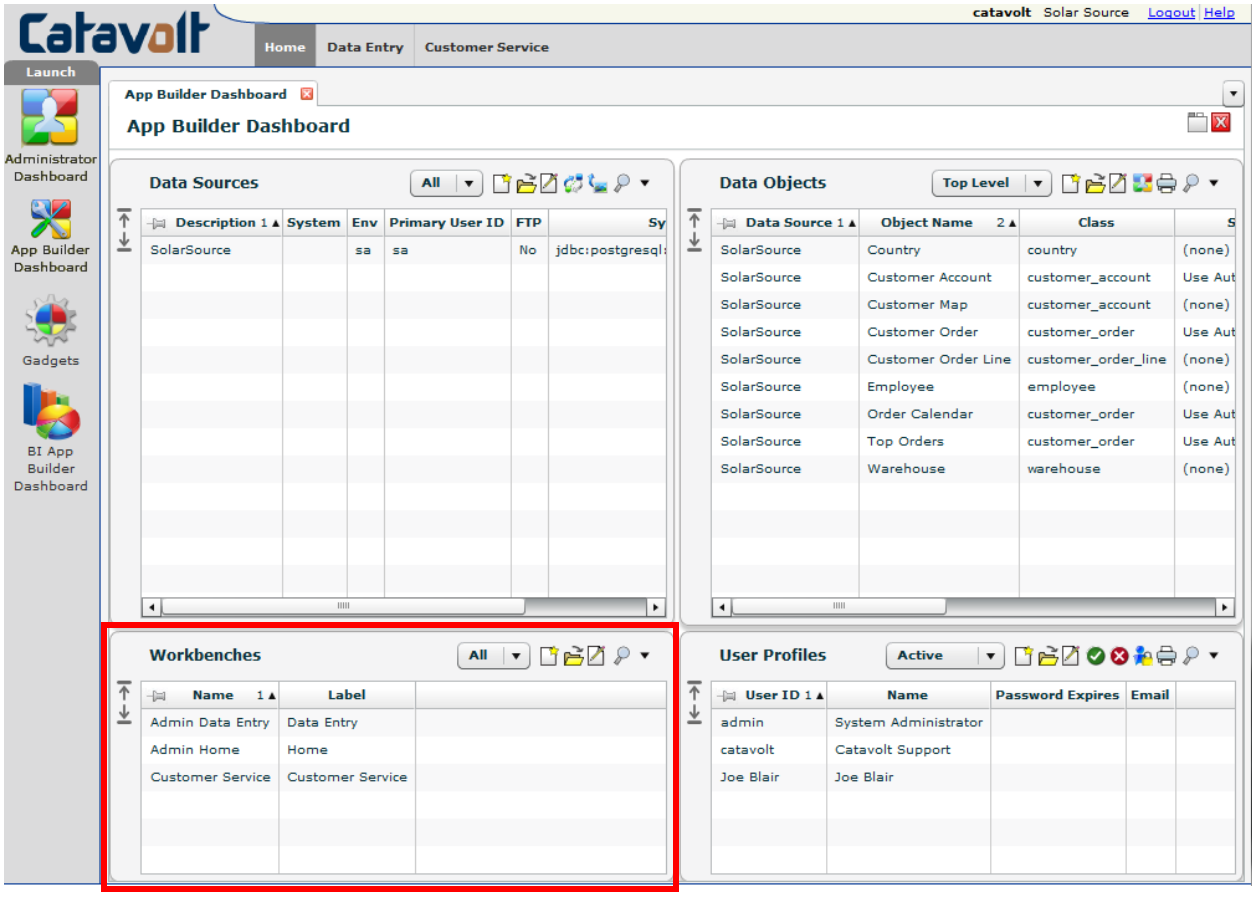


Figure 1: Xalt | Mobility in a web browser showing the workbenches in the top ribbon with the Customer Server workbench selected and the launchers on the left of the screen.

Accessing Workbenches

When a developer signs in to Xalt | Mobility, they will typically be immediately brought to the App Builder Dashboard. This dashboard is the hub that allows you to create and deploy Xalt applications. The Workbenches section shows you a list of all Workbenches defined in Xalt sorted by Name.



App Builder Dashboard

Data Sources

Description 1 ▲	System	Env	Primary User ID	FTP	Sy
SolarSource		sa	sa	No	jdbc:postgresql:

Data Objects

Data Source 1 ▲	Object Name 2 ▲	Class	S
SolarSource	Country	country	(none)
SolarSource	Customer Account	customer_account	Use Aut
SolarSource	Customer Map	customer_account	(none)
SolarSource	Customer Order	customer_order	Use Aut
SolarSource	Customer Order Line	customer_order_line	(none)
SolarSource	Employee	employee	(none)
SolarSource	Order Calendar	customer_order	Use Aut
SolarSource	Top Orders	customer_order	Use Aut
SolarSource	Warehouse	warehouse	(none)

Workbenches

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

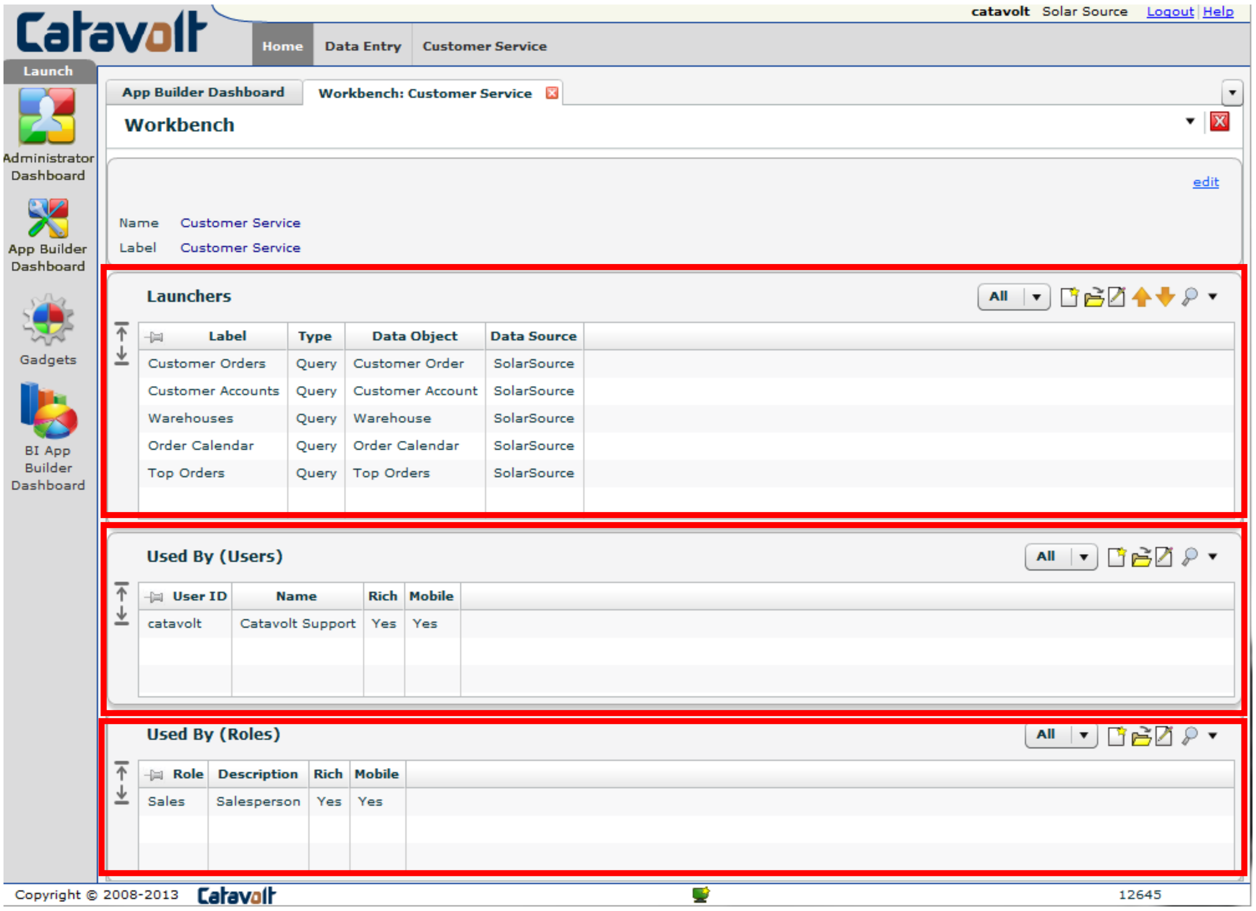
User Profiles

User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Figure 2: The App Builder Dashboard view with the Workbenches query section highlighted.

Workbench Components

A Workbench is made up of 3 components: Launchers (a list of Launchers on a Workbench), Used By (Users) (a list of end users who can access this Workbench), and Used By (Roles) (a list of security roles who can access this Workbench).



Workbench

Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

Used By (Users)

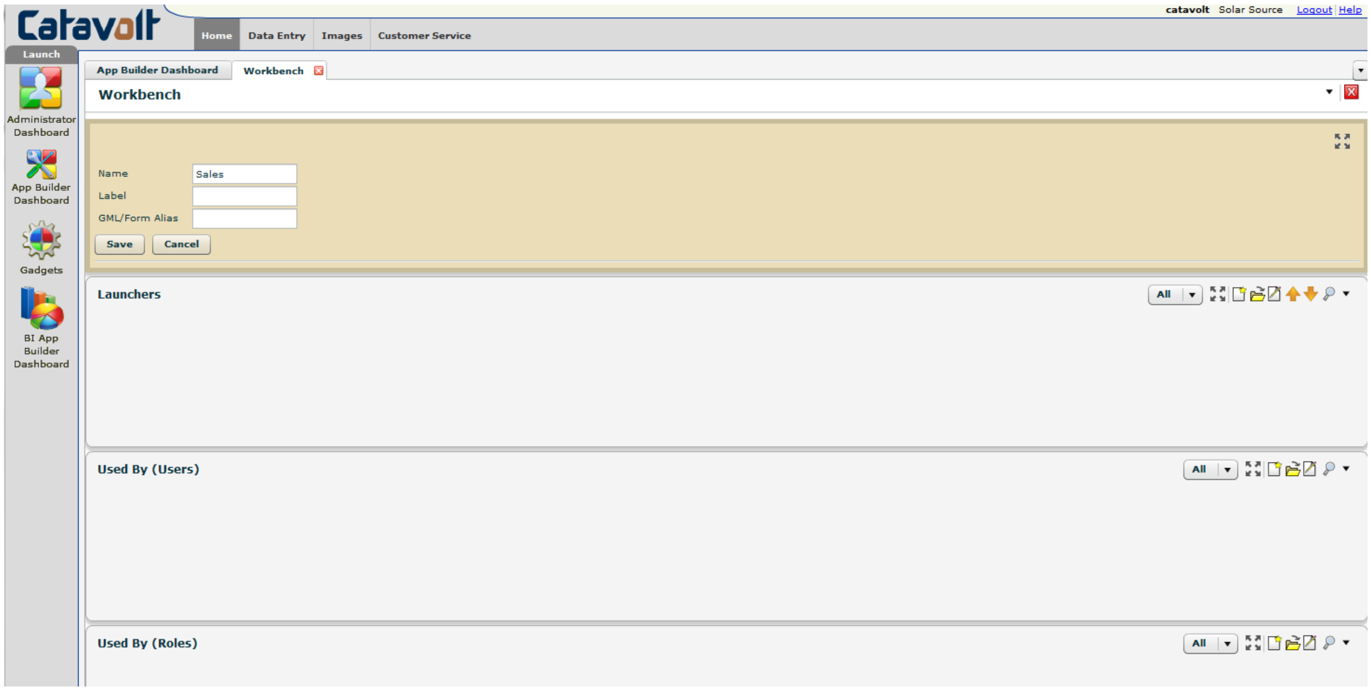
User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes

Used By (Roles)

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 3: The Workbench detail view with the Launchers and Used By query sections highlighted.

Creating Workbenches



Workbench

Name: Sales
Label:
GML/Form Alias:

Figure 4: The create Workbench details view.

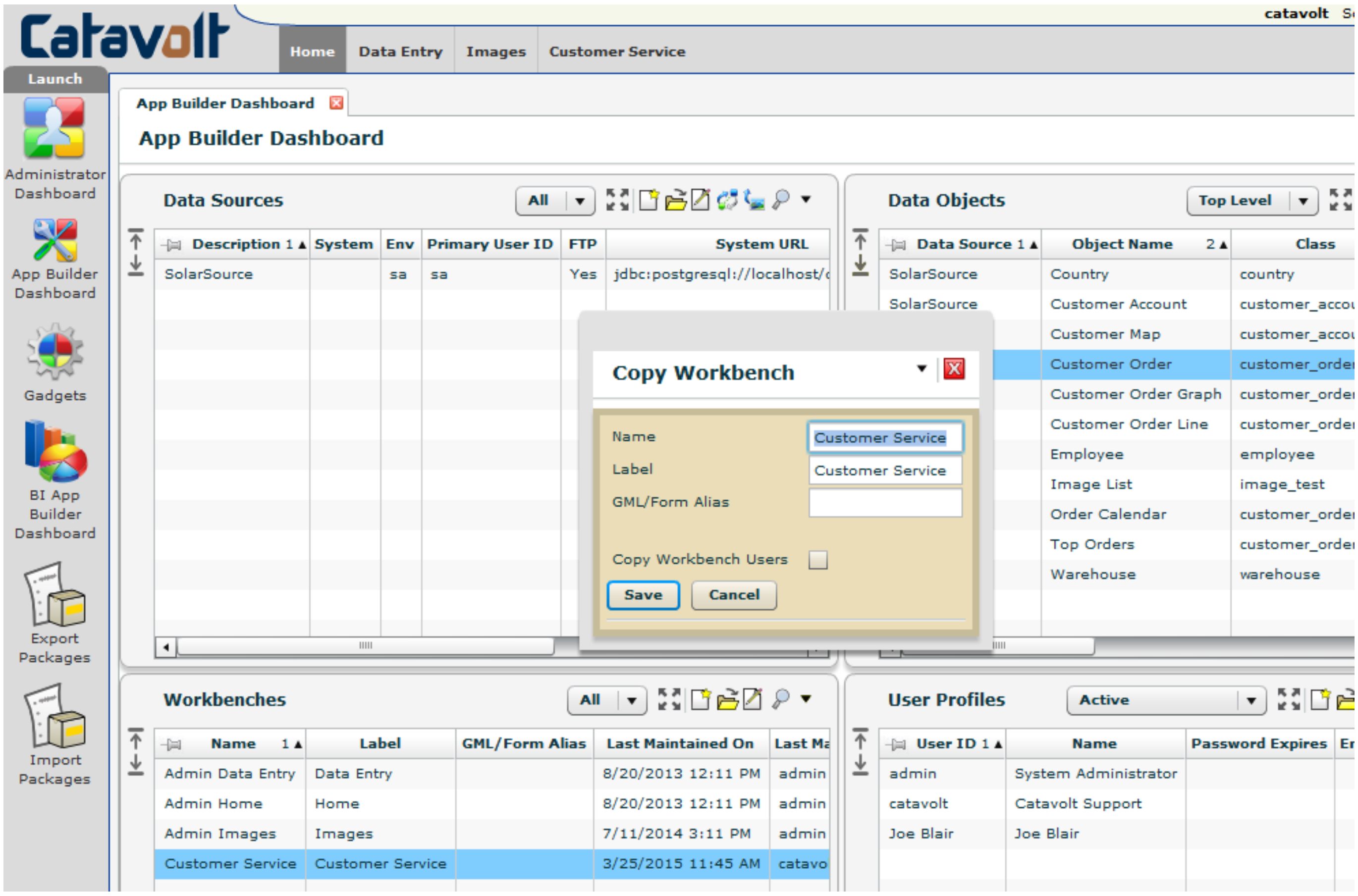
When creating a Workbench, you must specify the Workbench Name. This value is used to uniquely identify the Workbench to the system and will not be displayed to the user.

Label specifies the name that the end user will see when the Workbench is displayed. This value will default to the Name if it is not specified.

GML/Form Alias allows you to specify an alias when using this Action with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Copying Workbenches

There may be instances where you need to make a copy of a Workbench. You can select the Copy menu option to accomplish this. When Copying a Workbench, you will be prompted to supply the new Name and Label. An exact copy of this Workbench along with all of its launchers will be made. If you select the Copy Workbench Users option, all users assigned to the original Workbench will automatically be assigned to the new Workbench.



App Builder Dashboard

Data Sources

Description 1 ▲	System	Env	Primary User ID	FTP	System URL
SolarSource		sa	sa	Yes	jdbc:postgresql://localhost/(

Data Objects

Data Source 1 ▲	Object Name 2 ▲	Class
SolarSource	Country	country
SolarSource	Customer Account	customer_accou
	Customer Map	customer_accou
	Customer Order	customer_orde
	Customer Order Graph	customer_orde
	Customer Order Line	customer_orde
	Employee	employee
	Image List	image_test
	Order Calendar	customer_orde
	Top Orders	customer_orde
	Warehouse	warehouse

Copy Workbench

Name	Customer Service
Label	Customer Service
GML/Form Alias	
Copy Workbench Users	☐

Save Cancel

Workbenches

Name 1 ▲	Label	GML/Form Alias	Last Maintained On	Last Ma
Admin Data Entry	Data Entry		8/20/2013 12:11 PM	admin
Admin Home	Home		8/20/2013 12:11 PM	admin
Admin Images	Images		7/11/2014 3:11 PM	admin
Customer Service	Customer Service		3/25/2015 11:45 AM	catavo

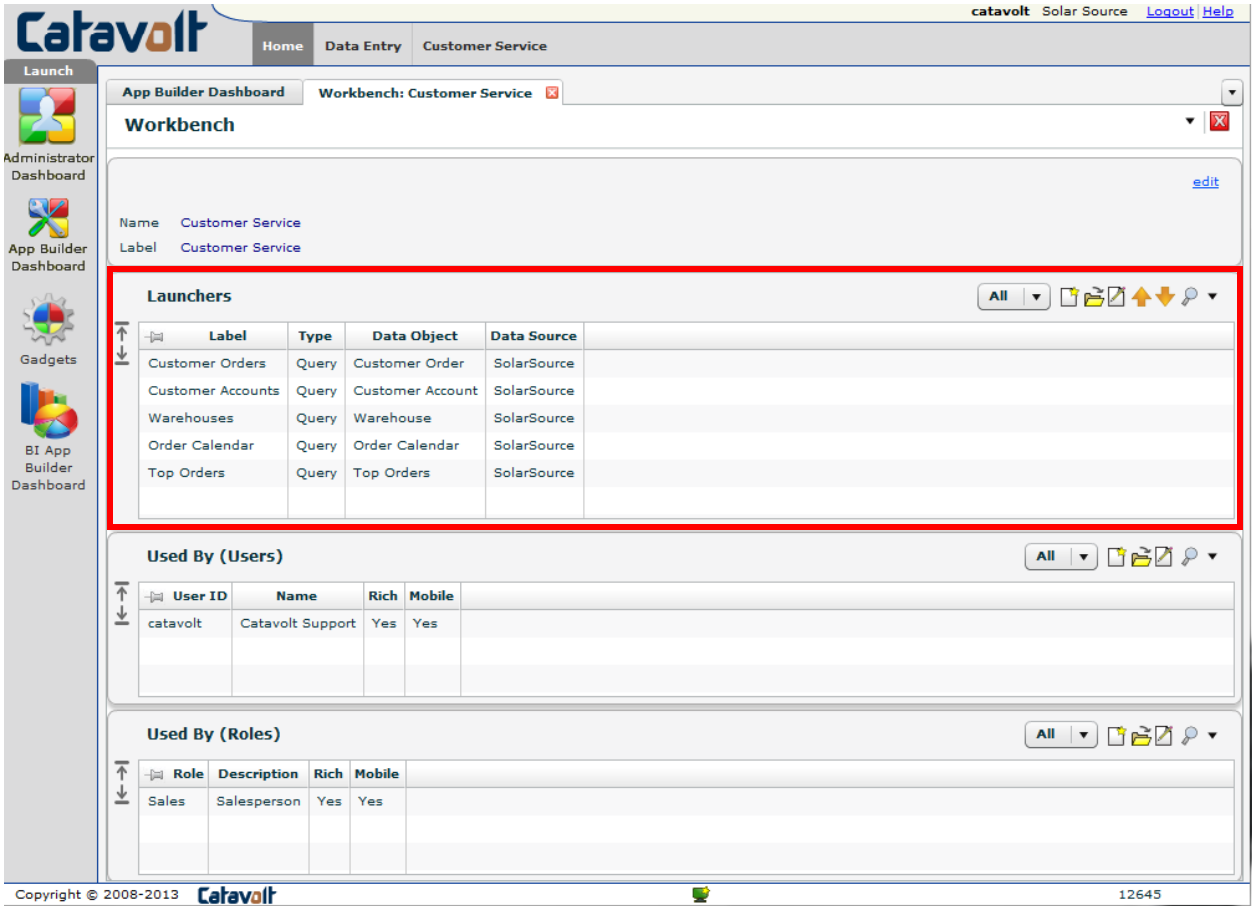
User Profiles

User ID 1 ▲	Name	Password Expires	Er
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Figure 5: The Copy Workbench prompt.

Launchers

The Launchers query section specifies which Launchers should be included with the Workbench and what in order they should appear. Multiple Launchers can be created for a Workbench. They will appear in the Launchers list to the end user in the same order they appear below. To change the order, select a Launcher and press the Move Up and Move Down toolbar buttons.



Workbench

edit

Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

Used By (Users)

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes

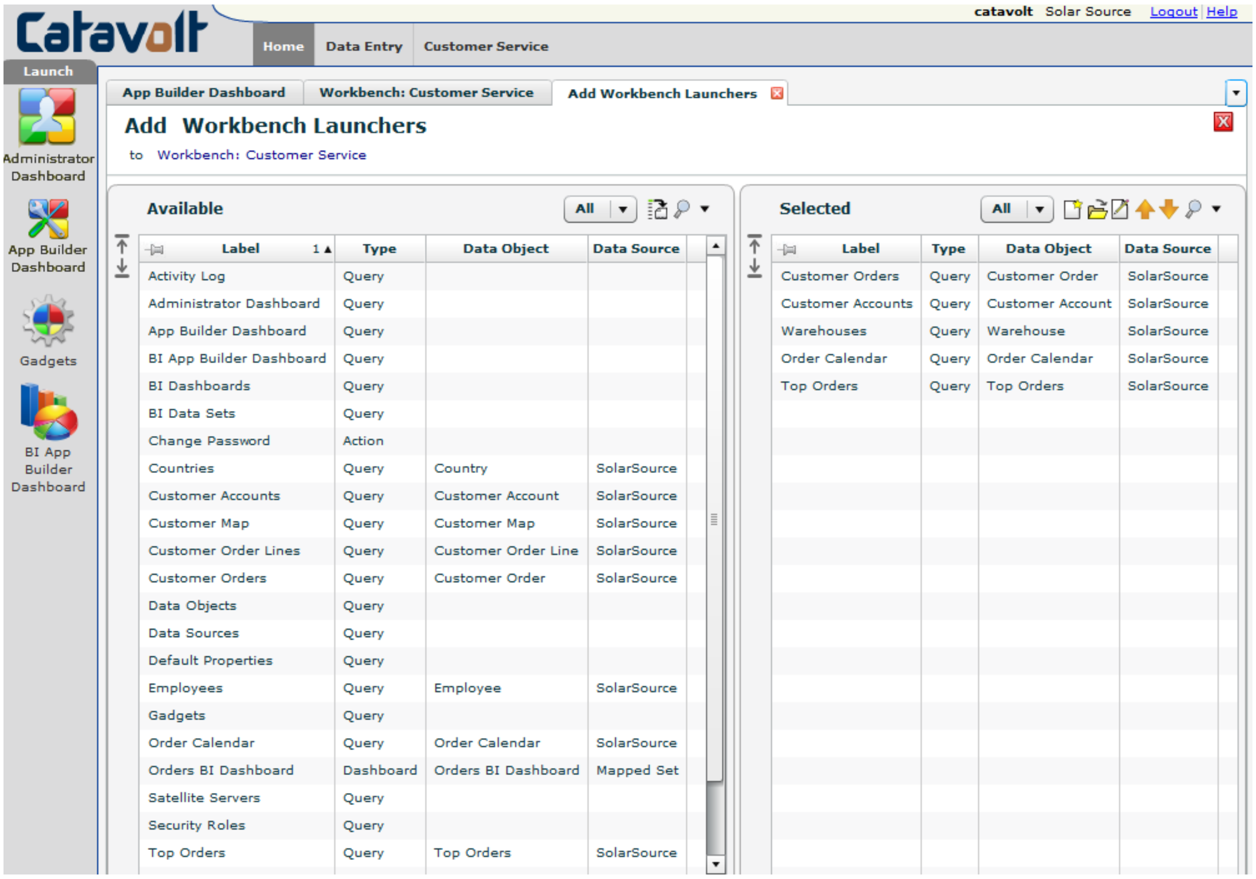
Used By (Roles)

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Copyright © 2008-2013 Catavolt 12645

Figure 6: The Workbench details view with the Launchers query section highlighted.

When adding Launchers, you will be presented with two lists. The Available Launchers list shows all launchers. The Selected Launchers list shows the launchers that will be displayed on the workbench.



Add Workbench Launchers

to Workbench: Customer Service

Available

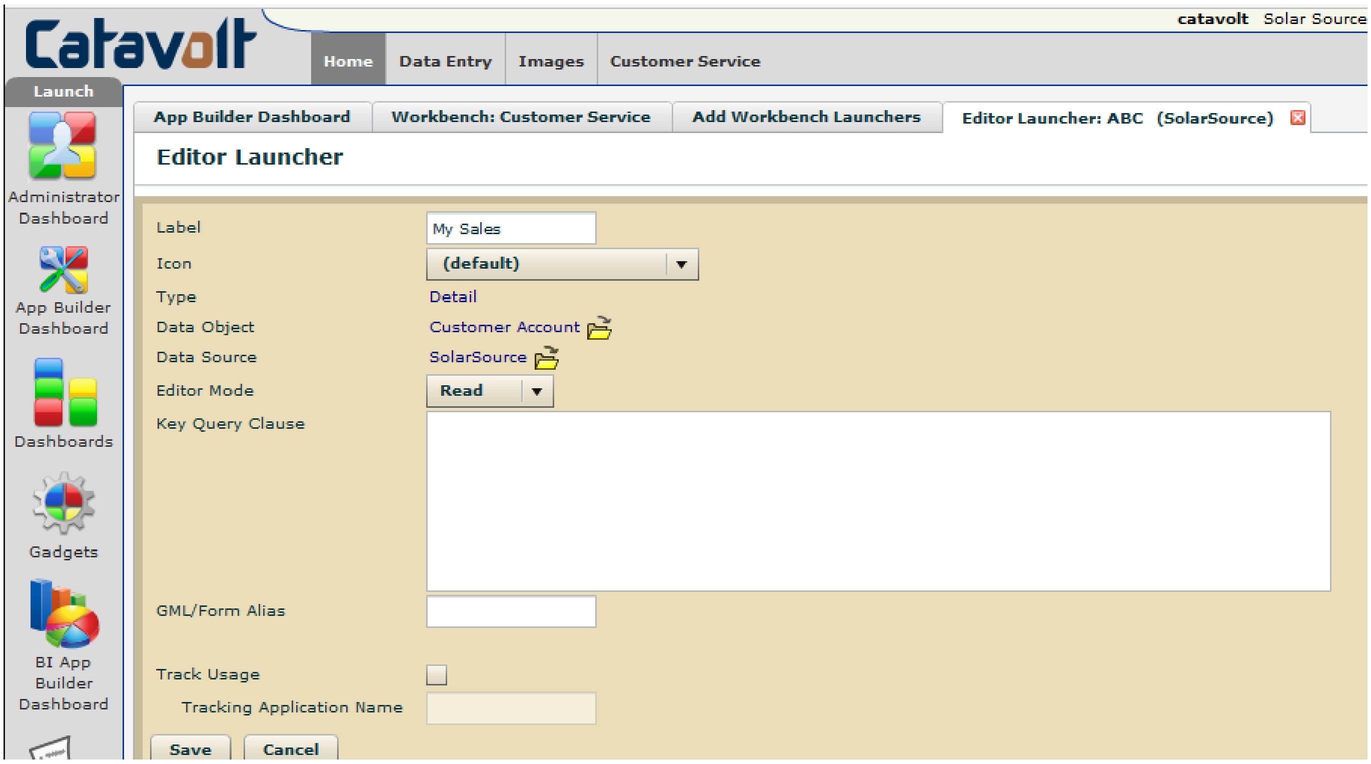
Label	Type	Data Object	Data Source
Activity Log	Query		
Administrator Dashboard	Query		
App Builder Dashboard	Query		
BI App Builder Dashboard	Query		
BI Dashboards	Query		
BI Data Sets	Query		
Change Password	Action		
Countries	Query	Country	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Customer Map	Query	Customer Map	SolarSource
Customer Order Lines	Query	Customer Order Line	SolarSource
Customer Orders	Query	Customer Order	SolarSource
Data Objects	Query		
Data Sources	Query		
Default Properties	Query		
Employees	Query	Employee	SolarSource
Gadgets	Query		
Order Calendar	Query	Order Calendar	SolarSource
Orders BI Dashboard	Dashboard	Orders BI Dashboard	Mapped Set
Satellite Servers	Query		
Security Roles	Query		
Top Orders	Query	Top Orders	SolarSource

Selected

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

Figure 7: The Add Workbench Launchers view.

You can select a single or multiple launchers and press the Add button to add them to the Workbench. After the Workbench Launcher has been created, opening the Launcher record will display the details for the Launcher.



Editor Launcher

Label	My Sales
Icon	(default)
Type	Detail
Data Object	Customer Account
Data Source	SolarSource
Editor Mode	Read
Key Query Clause	
GML/Form Alias	
Track Usage	□
Tracking Application Name	

Figure 8: The details view for an Editor Launcher.

Label specifies the text that will appear on the Launcher.

Icon specifies the image that will appear on the Launcher.

Type specifies what the Launcher will do when clicked. This value is set automatically and cannot be changed. The allowable values are:

- Query – Display a list of records
- Detail – Display a single record (See Chapter 3: Create Dashboard Launcher for more information on creating a Detail Launcher).
- Action – Performs an Action (See Chapter 6: Creating Actions for more information on how an Action Launcher is created).
- BI Dashboard – Display a BI Dashboard (See Chapter 11: Business Intelligence (BI) for more information on how a BI Dashboard Launcher is created).
- Dashboard – Display a Dashboard (See Chapter 14: Dashboards for more information on how a Dashboard Launcher is created).
- Power BI – Display a Microsoft Power BI Report (See Chapter 2: Data Sources for more information on how a Power BI Data Source is created).

Data Object specifies the Data Object that this launcher will display records for. This value is set automatically and cannot be changed.

Data Source specifies the Data Source that this launcher will use to query data. This value is set automatically and cannot be changed.

Key Query Clause specifies the criteria that is used to determine which record in the list should be used in the detail. This value can only be changed for Launchers of type Detail. Note that the query clause specified must return exactly 1 record. If 0 or more than 1 record is returned by the query clause, an error will be displayed when running the launcher. If the detail needs to show a different record for each user that runs it, you should use User Properties in the query clause to tailor it to each user. See Appendix B: Specifying Where Clauses for more information about specifying Additional Where Clause values.

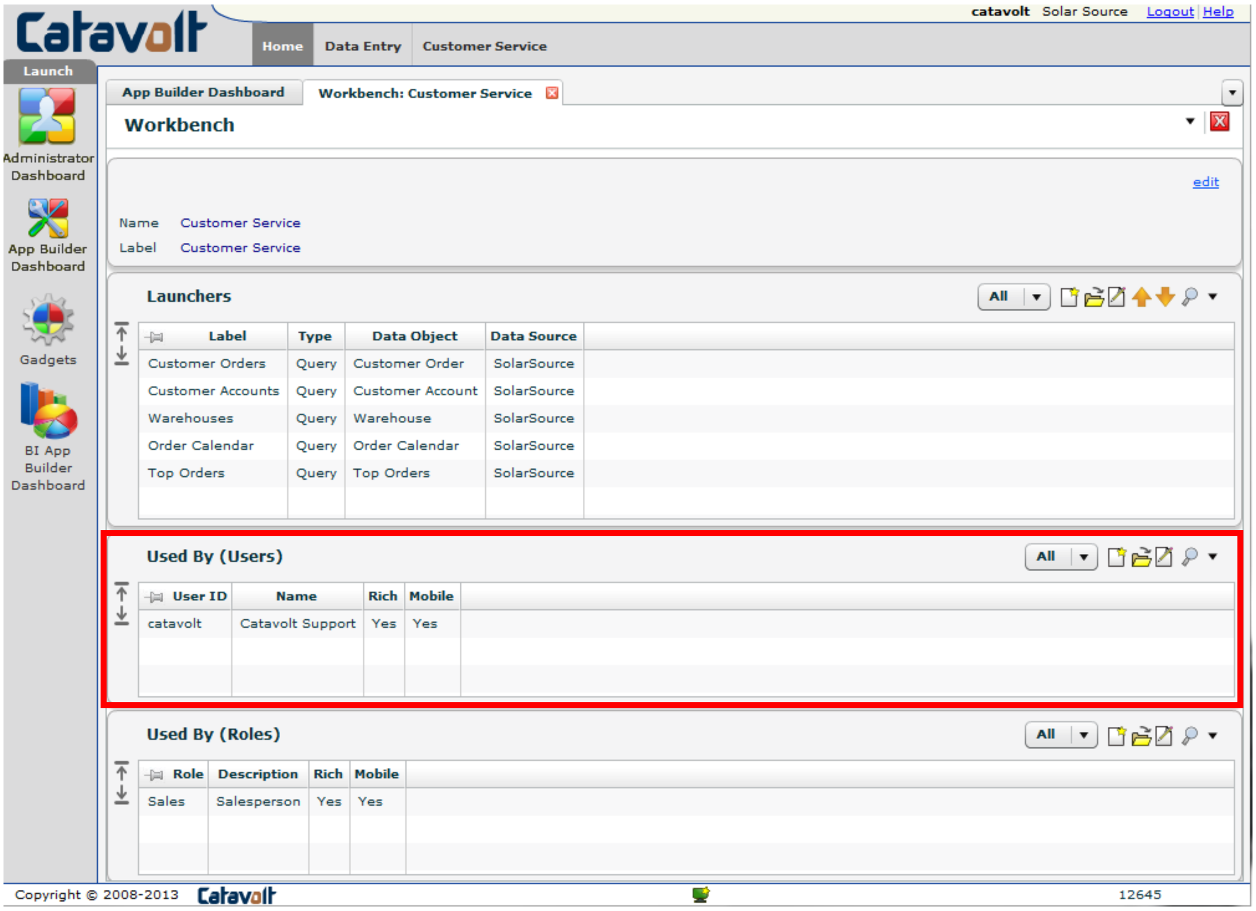
GML/ Form Alias allows you to specify an alias when using this Action with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Track Usage allows you to track when users run this launcher. Every time a user runs this launcher, a record will be written to a tracking table that contains information about the launcher, the user running it, and the date and time it was run. Note that this option is only available if you are licensed to the Xalt Metrics application.

Tracking Application Name specifies the name of the application to use when tracking. By default, the application name will be the name of the launcher. This can be changed to allow multiple launchers to be grouped together under the same application for reporting purposes. Note that this option is only available if you are licensed to the Xalt Metrics application.

Used By (Users)

Used By (Users) specifies the list of end users who can access this Workbench.



Workbench

Name Customer Service

Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

Used By (Users)

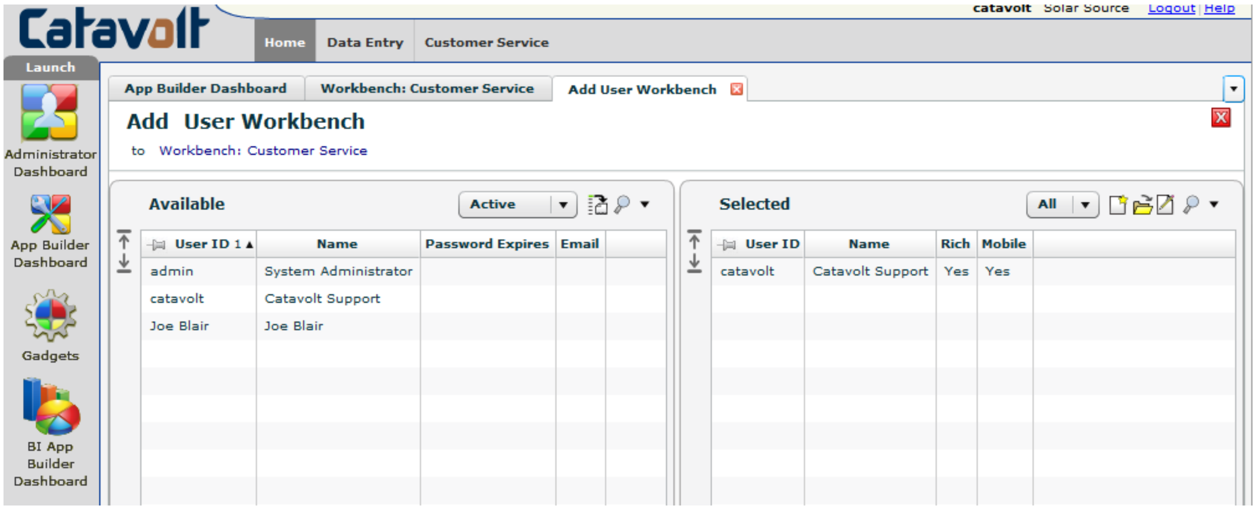
User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes

Used By (Roles)

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 9: The Workbench details view with the Used By query section highlighted.

When adding Workbench Users, you will be presented with two lists. The Available Users list shows all Hexagon users. The Selected users list shows the user that will be running the workbench.



Add User Workbench

to Workbench: Customer Service

Available

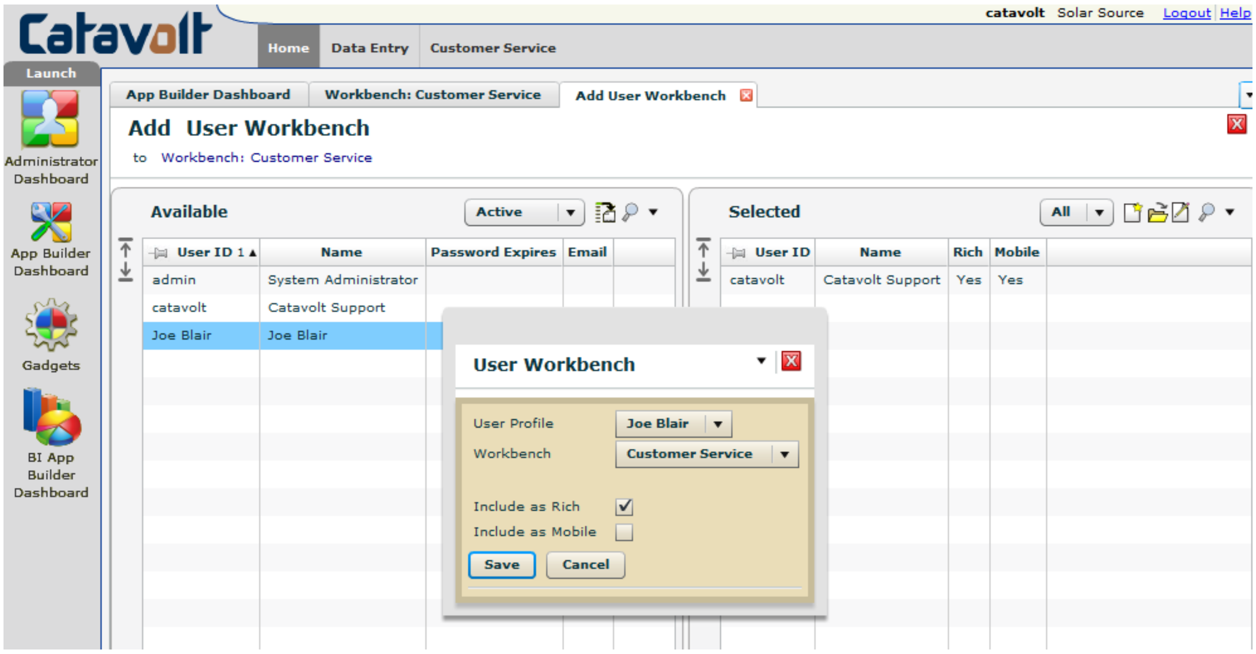
User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Selected

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes

Figure 10: The add User Workbench view.

You can select a single or multiple users and press the Add button to add them to the Workbench. If you select a single user, you will be presented with the following dialog:



Add User Workbench

to Workbench: Customer Service

Available

User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Selected

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes

User Workbench

User Profile: Joe Blair

Workbench: Customer Service

Include as Rich ☑

Include as Mobile ☐

Figure 11: The add User Workbench dialog.

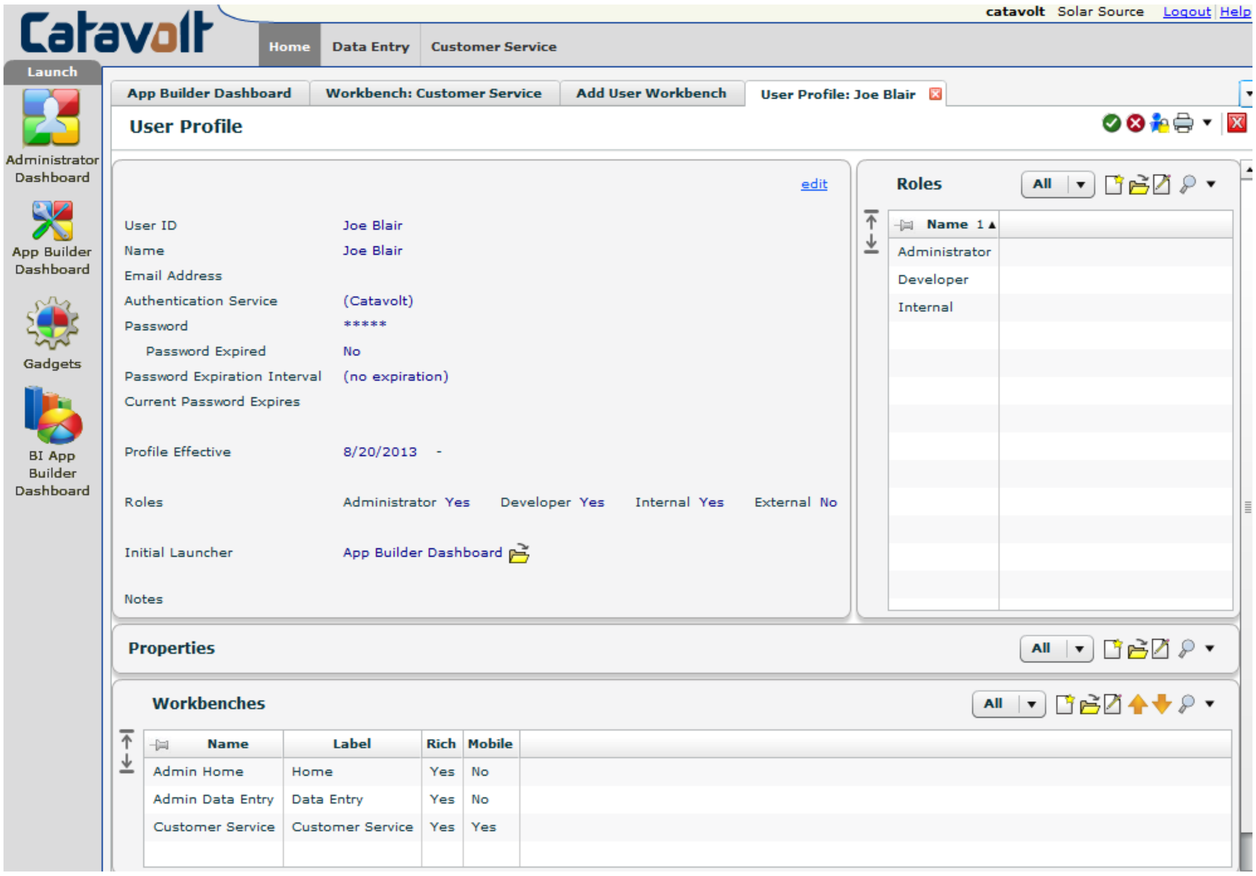
When creating a User Workbench, User Profile specifies the user that is to be granted access to the Workbench. User Profile displays a list of all users in the system.

Workbench specifies the Workbench to which the User Profile is to be added. Workbench displays a list of all Workbenches in the system. The current Workbench you are working with will be the default selected value.

Include as Rich specifies whether the user should have access to this workbench when running the rich (web) client.

Include as Mobile specifies whether the user should have access to this workbench when running the mobile client.

After the User Workbench has been created, opening the Used By record will display the details for the User. See Chapter 9: User Profile Administration for more information about User Profiles.



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Notes	

Roles

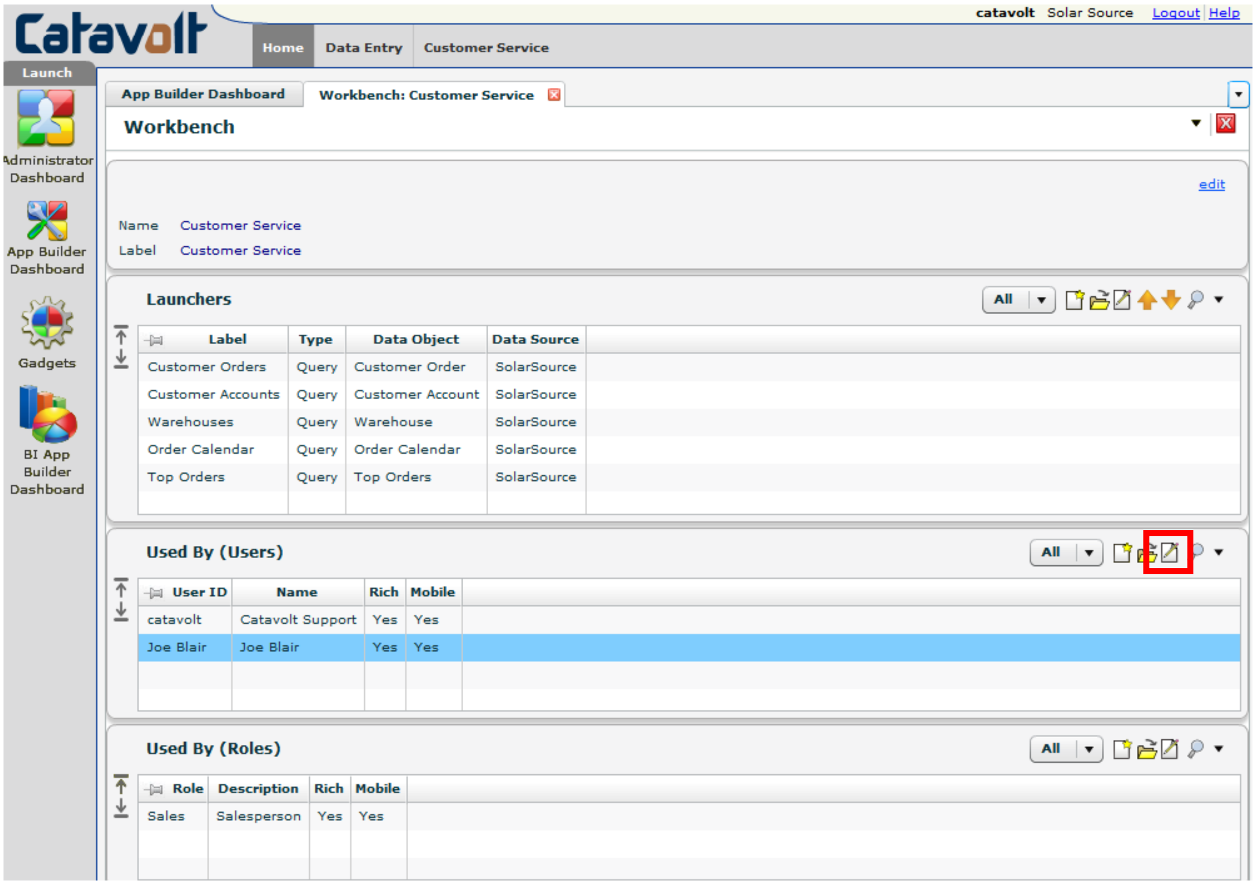
Name 1 ▲
Administrator
Developer
Internal

Workbenches

Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

Figure 12: The User Profile details view.

If you need to change the Include as Rich or Include as Mobile value after creating the User Workbench, select the User Workbench record and choose the Edit Action



Workbench

Name Customer Service

Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

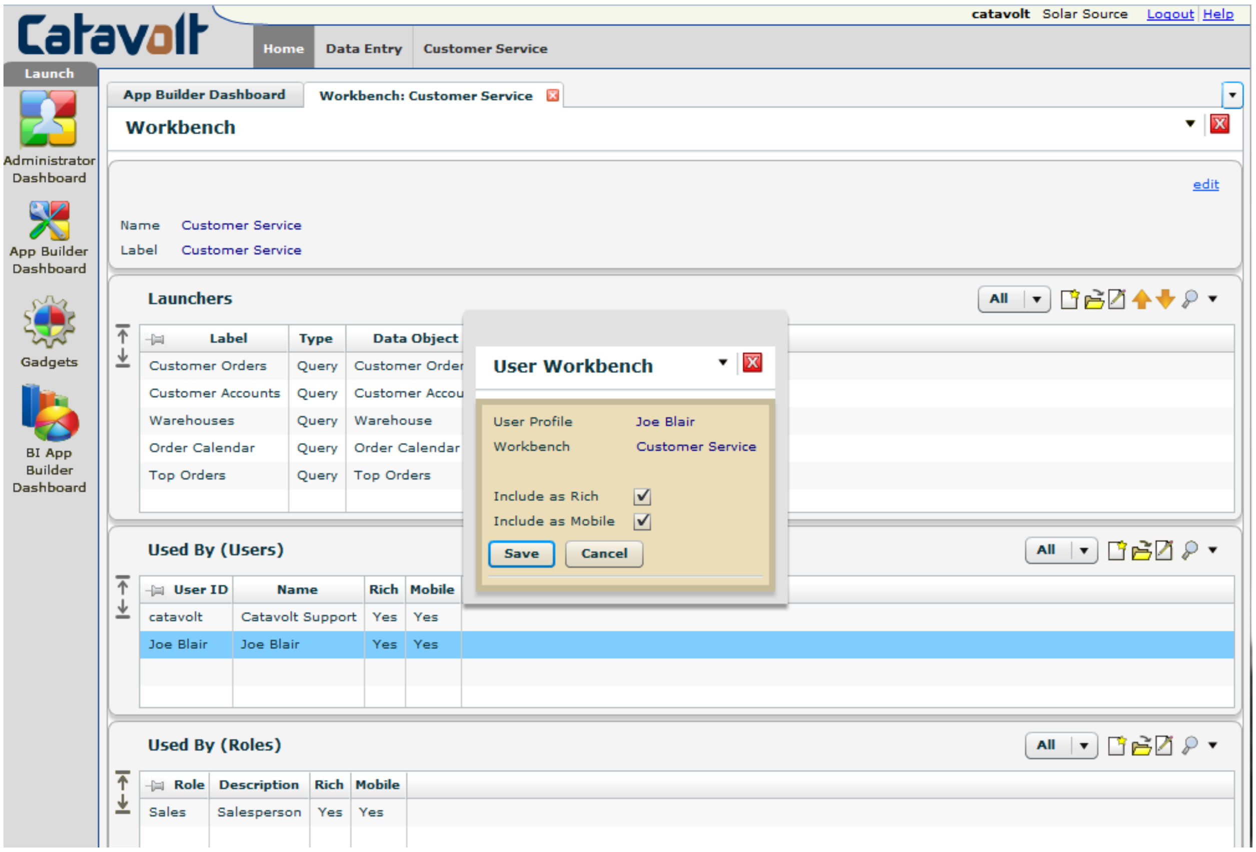
Used By (Users)

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes
Joe Blair	Joe Blair	Yes	Yes

Used By (Roles)

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 13: Changing the Include as Rich and Include as Mobile settings.



Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object
Customer Orders	Query	Customer Order
Customer Accounts	Query	Customer Accou
Warehouses	Query	Warehouse
Order Calendar	Query	Order Calendar
Top Orders	Query	Top Orders

User Workbench

User Profile Joe Blair
Workbench Customer Service

Include as Rich ☑
Include as Mobile ☑

Used By (Users)

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes
Joe Blair	Joe Blair	Yes	Yes

Used By (Roles)

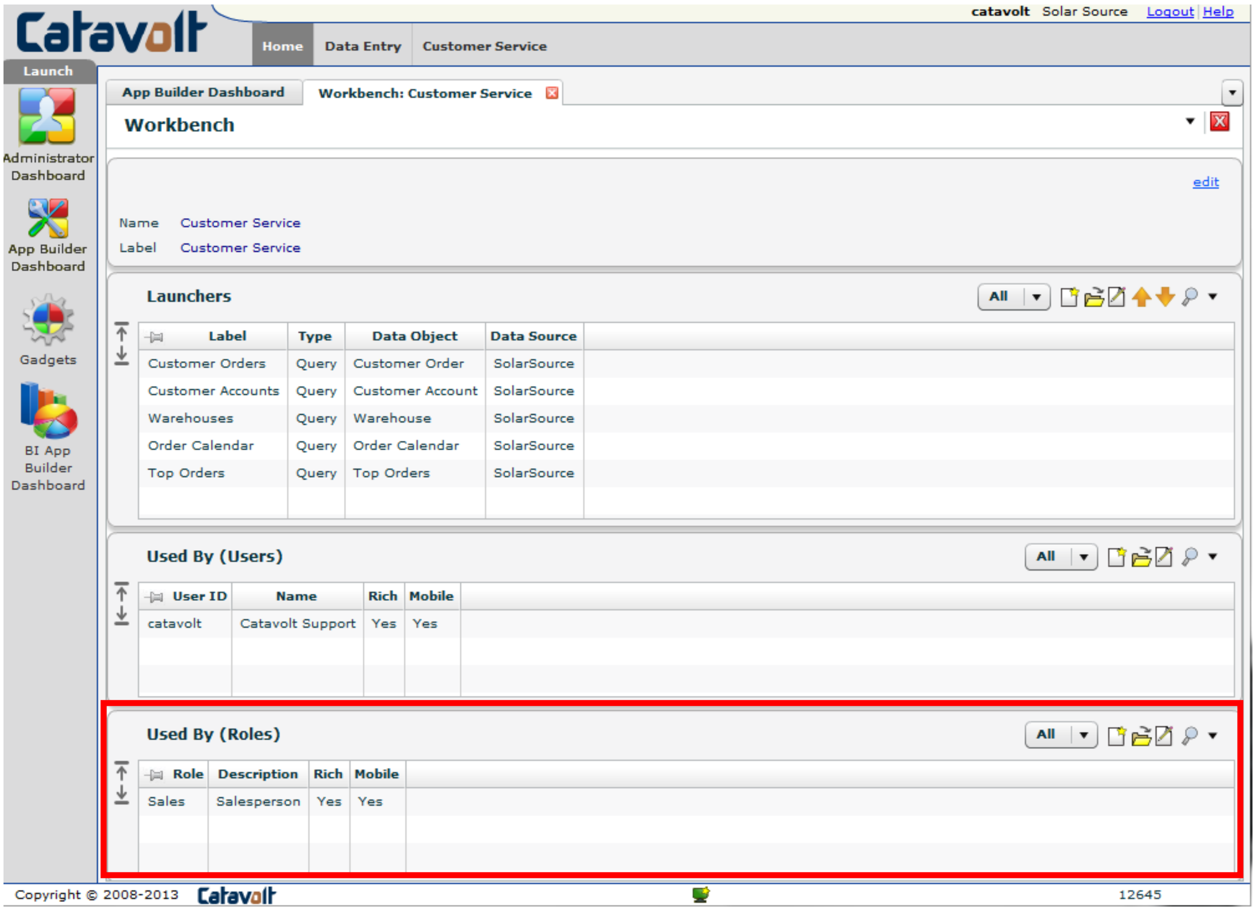
Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 14: Changing the Include as Rich and Include as Mobile settings.

Used By (Roles)

Used By (Roles) specifies the list of Security Roles who can access this Workbench. User Profiles that do not have Workbenches defined on the profile itself will automatically pick up Workbenches from the Security Roles that they are assigned to. You can alter the order the Workbenches appear by arranging the list of Security Roles on the User Profile as well as arranging the list of Workbenches on the Security Profile. Note that if a Workbench exists on multiple Security Roles, the first instance will be used (duplicate appearances will be removed).

This feature is also available for users who have been authenticated remotely and have had Security Roles dynamically added during the authentication process. Remote authentication is beyond the scope of this document. Please contact Hexagon Support to request more information on this topic.



Workbench

Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

Used By (Users)

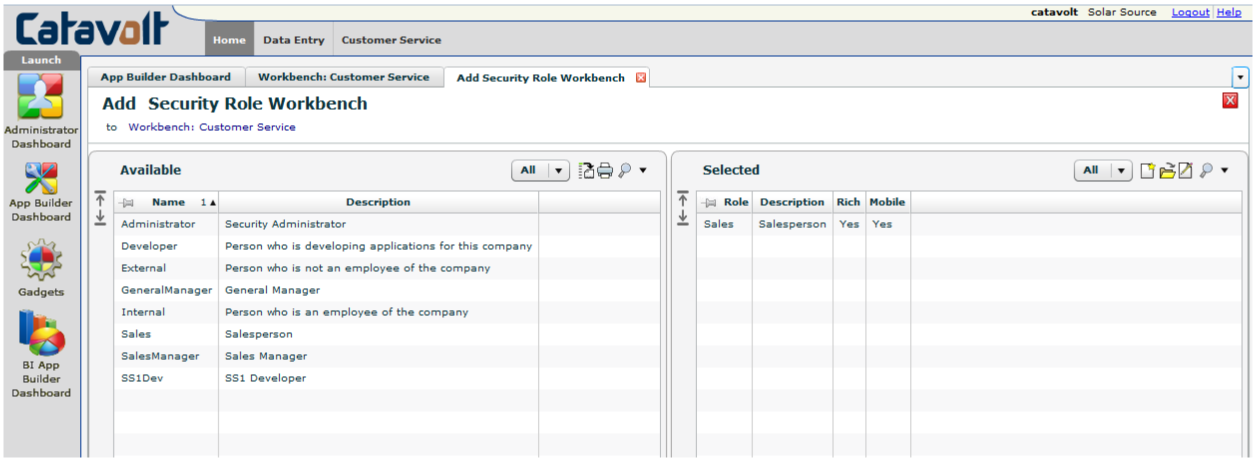
User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes

Used By (Roles)

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 15: The Workbench details view with the Used By query section highlighted.

When adding Workbench Security Roles, you will be presented with two lists. The Available Roles list shows all Security Roles. The Selected Roles list shows the roles that will be running the workbench.



Add Security Role Workbench

to Workbench: Customer Service

Available

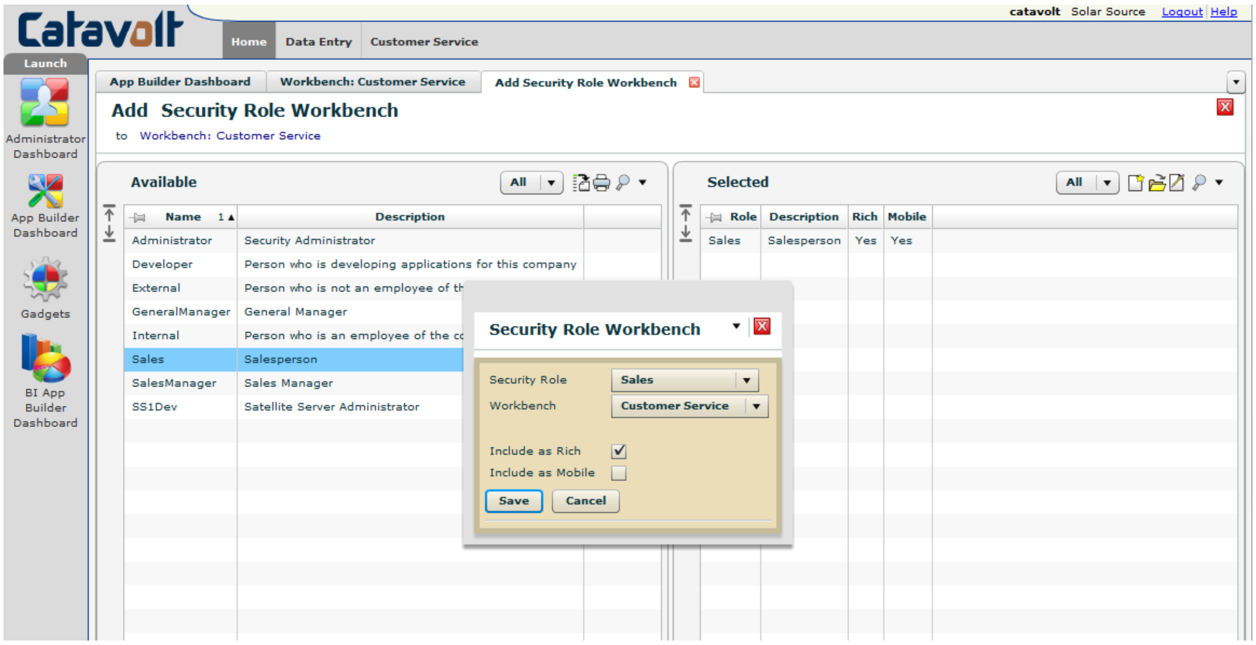
Name	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	SS1 Developer

Selected

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 16: The add Security Role Workbench view.

You can select a single or multiple Security Roles and press the Add button to add them to the Workbench. If you select a single role, you will be presented with the following dialog:



Add Security Role Workbench

to Workbench: Customer Service

Available

Name	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of th
GeneralManager	General Manager
Internal	Person who is an employee of the cc
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	Satellite Server Administrator

Selected

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Security Role Workbench

Security Role: Sales
Workbench: Customer Service
Include as Rich ☑
Include as Mobile ☐
Save Cancel

Figure 17: The add Security Role Workbench dialog.

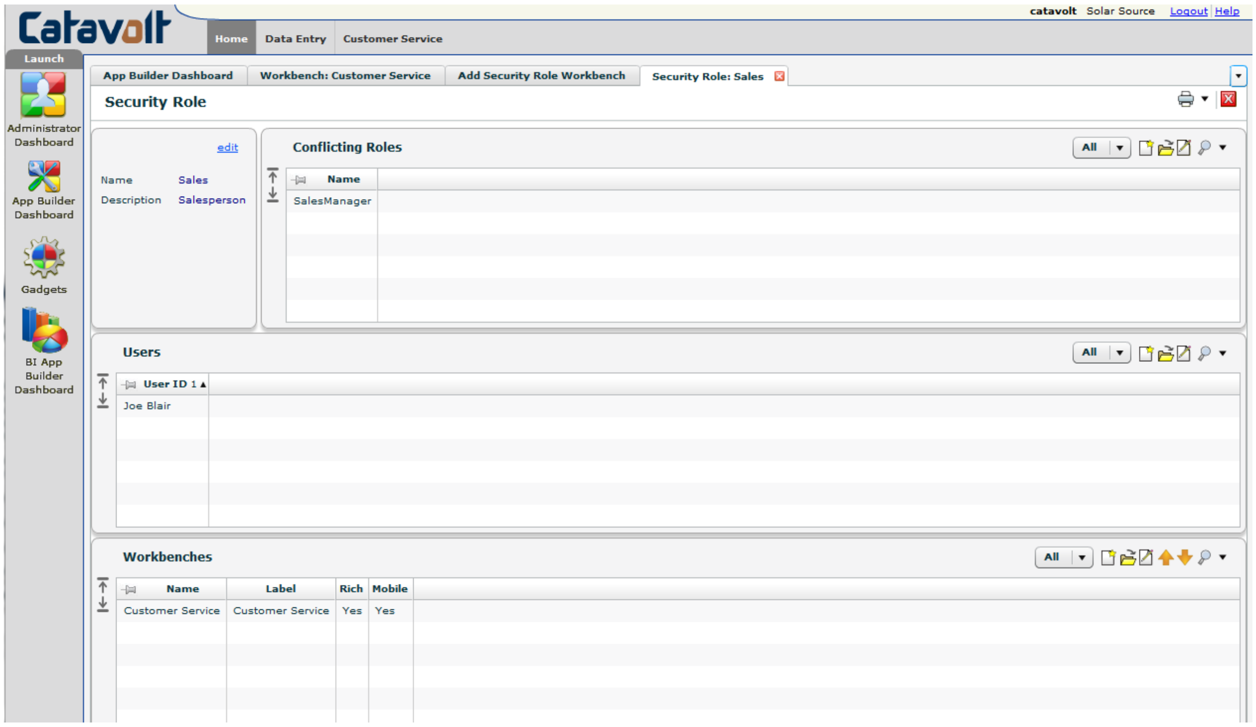
When creating a User Workbench, Security Role specifies the role that is to be granted access to the Workbench. Security Role displays a list of all roles in the system.

Workbench specifies the Workbench to which the Security Role is to be added. Workbench displays a list of all Workbenches in the system. The current Workbench you are working with will be the default selected value.

Include as Rich specifies whether the role should have access to this workbench when running the rich (web) client.

Include as Mobile specifies whether the role should have access to this workbench when running the mobile client.

After the Security Role Workbench has been created, opening the Used By record will display the details for the Security Role. See Chapter 10: Security for more information about Security Roles.



Security Role

Name Sales
Description Salesperson

Conflicting Roles

Name
SalesManager

Users

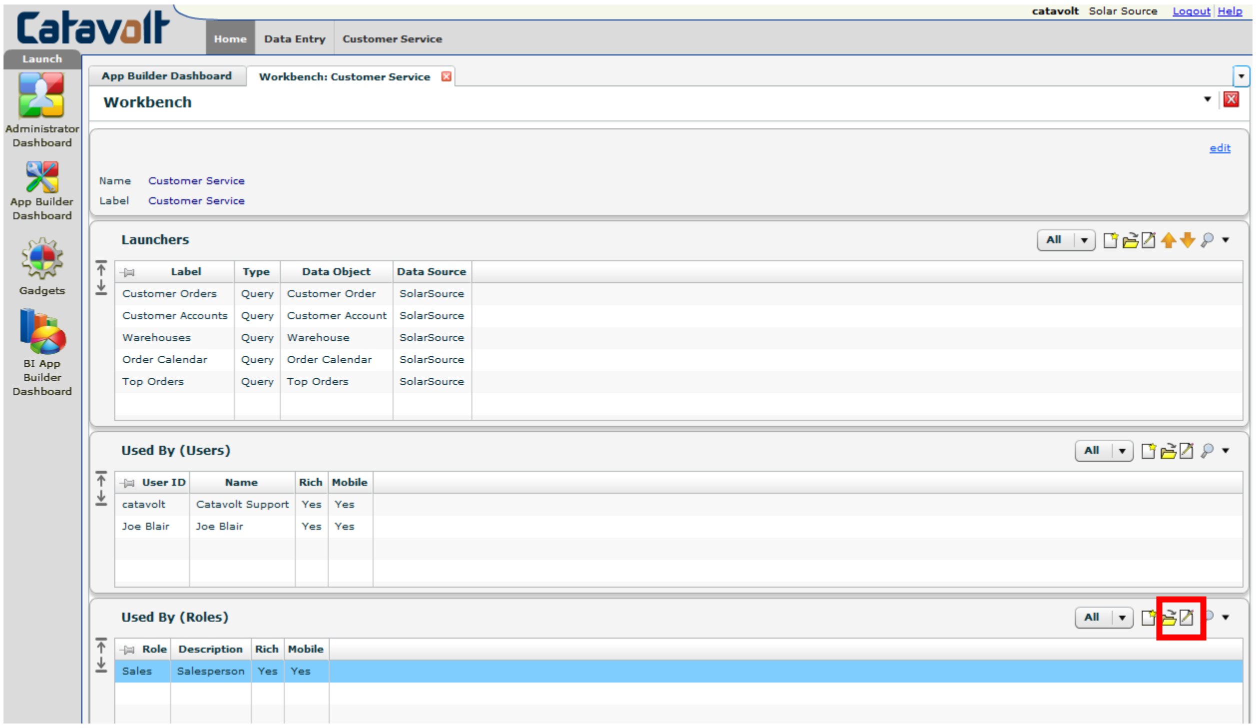
User ID 1 ▲
Joe Blair

Workbenches

Name	Label	Rich	Mobile
Customer Service	Customer Service	Yes	Yes

Figure 18: The Security Role details view.

If you need to change the Include as Rich or Include as Mobile value after creating the Security Role Workbench, select the Security Role Workbench record and choose the Edit Action



Workbench

Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

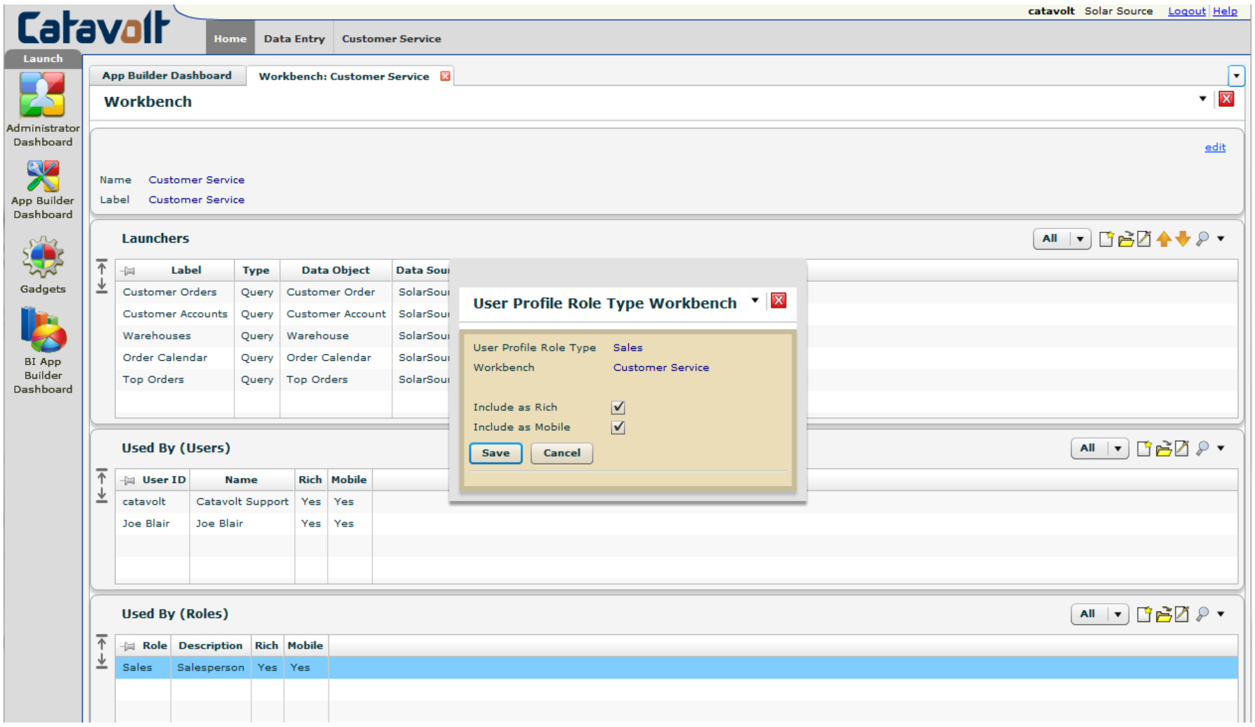
Used By (Users)

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes
Joe Blair	Joe Blair	Yes	Yes

Used By (Roles)

Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 19: Changing the Include as Rich and Include as Mobile settings.



Workbench

Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object	Data Sou
Customer Orders	Query	Customer Order	SolarSou
Customer Accounts	Query	Customer Account	SolarSou
Warehouses	Query	Warehouse	SolarSou
Order Calendar	Query	Order Calendar	SolarSou
Top Orders	Query	Top Orders	SolarSou

User Profile Role Type Workbench

User Profile Role Type Sales
Workbench Customer Service

Include as Rich ☑
Include as Mobile ☑

Save Cancel

Used By (Users)

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes
Joe Blair	Joe Blair	Yes	Yes

Used By (Roles)

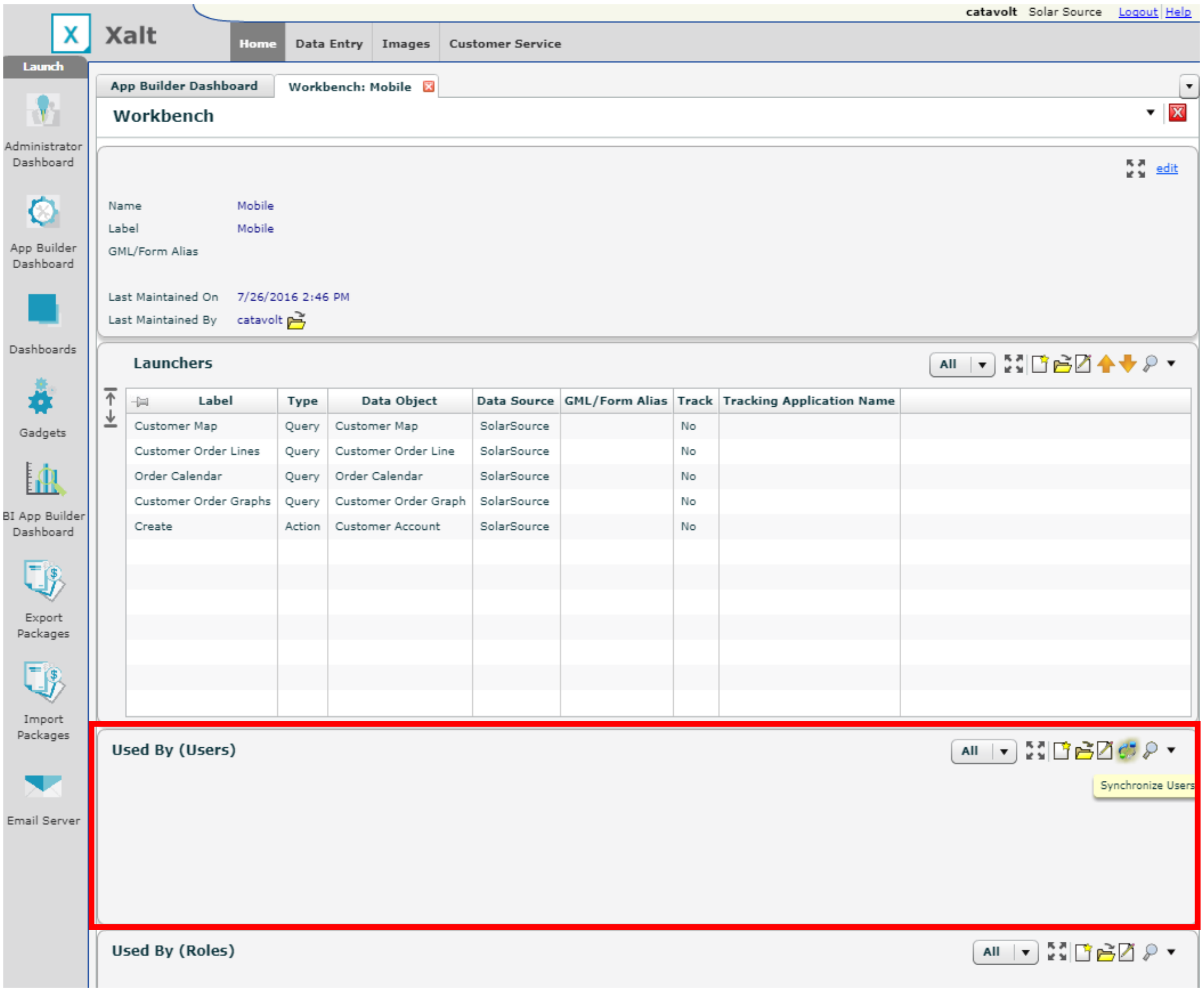
Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 20: Changing the Include as Rich and Include as Mobile settings.

Other Workbench Menu Options

Synchronize Users

In some cases, it is desirable to take all the users assigned to one Workbench and assign them to a second Workbench without having to add each user individually. This can happen if you create or import a new Workbench that may need to share a set of users with an existing Workbench. In order to accomplish this, we have added the Synchronize Users menu action on the Used By (Users) section of Workbench



Workbench

Name Mobile
Label Mobile
GML/Form Alias

Last Maintained On 7/26/2016 2:46 PM
Last Maintained By catavolt

Launchers

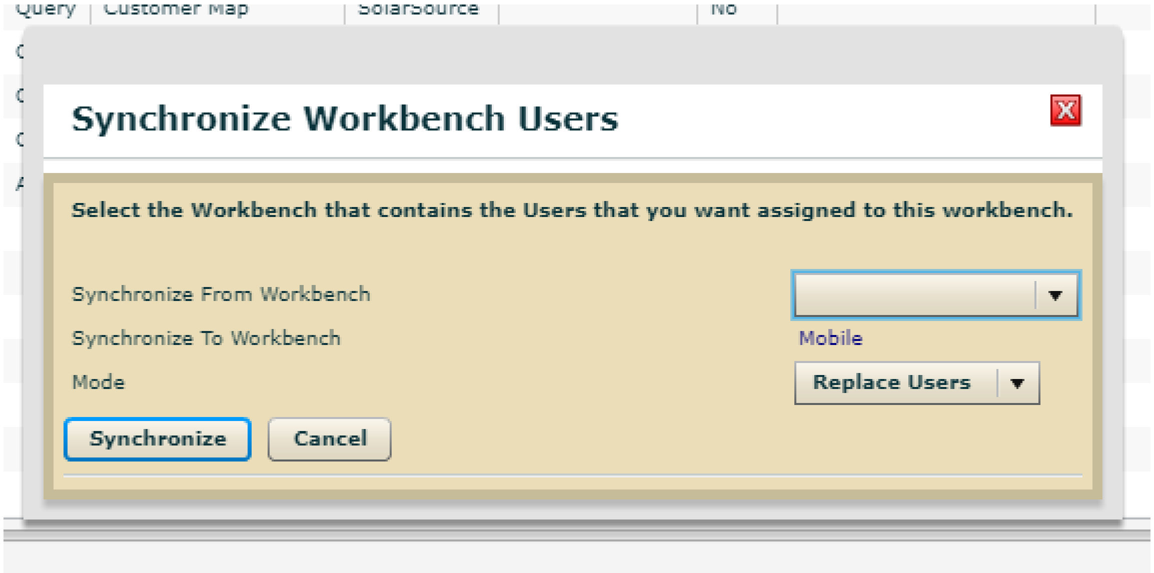
Label	Type	Data Object	Data Source	GML/Form Alias	Track	Tracking Application Name
Customer Map	Query	Customer Map	SolarSource		No	
Customer Order Lines	Query	Customer Order Line	SolarSource		No	
Order Calendar	Query	Order Calendar	SolarSource		No	
Customer Order Graphs	Query	Customer Order Graph	SolarSource		No	
Create	Action	Customer Account	SolarSource		No	

Used By (Users)

Used By (Roles)

Figure 21: Workbench Details with the Synchronize Users action highlighted.

Selecting this action will bring up the following prompt:



Synchronize Workbench Users

Select the Workbench that contains the Users that you want assigned to this workbench.

Synchronize From Workbench

Synchronize To Workbench: Mobile

Mode: Replace Users

Figure 22: Synchronize Workbench Users Details

Synchronize From Workbench specifies the Workbench to pull the list of assigned users from. Synchronize From Workbench displays a list of all Workbenches in the system (except the current Workbench you are working with).

Synchronize To Workbench specifies the Workbench to assign the new list of Users to. The current Workbench you are working with will be the default selected value.

Mode specifies how the list of users should be assigned. The allowable values are:

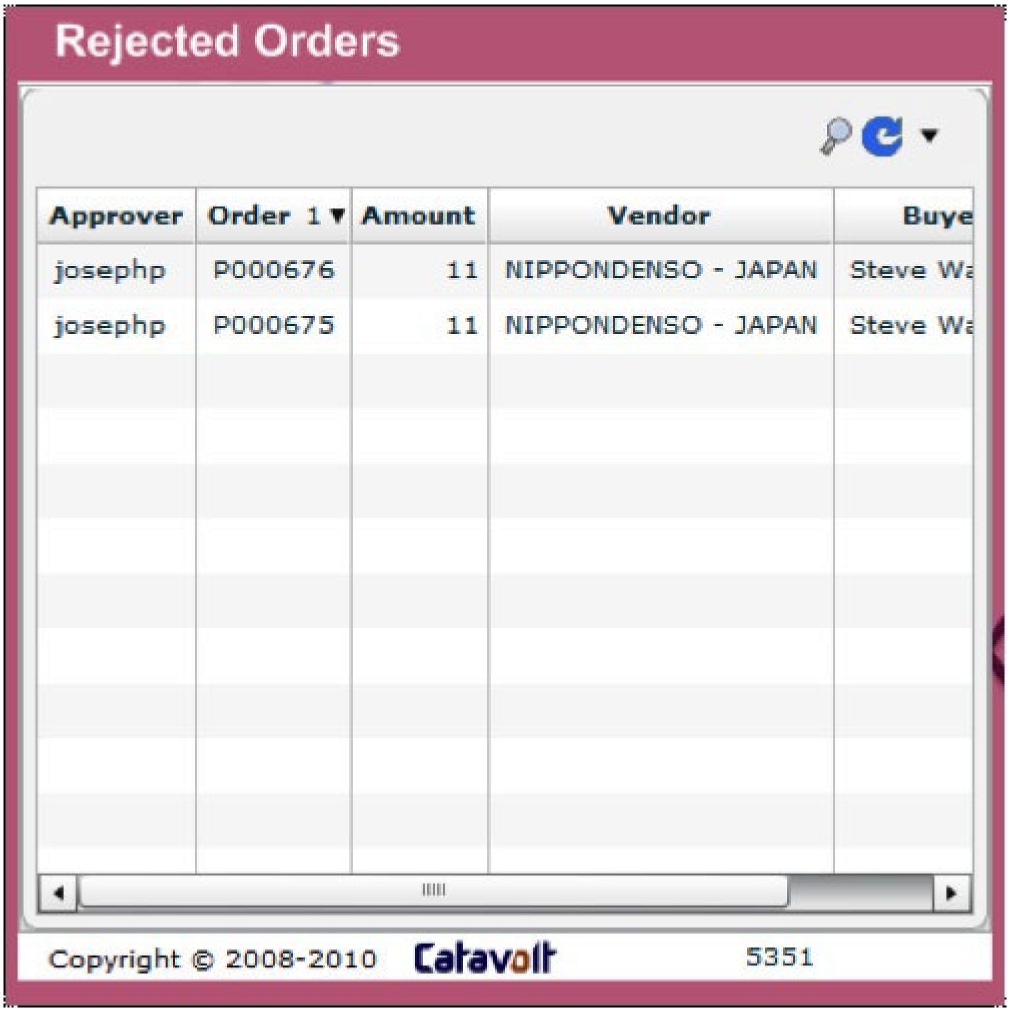
Replace Users – Remove all assigned users from this Workbench, then add all assigned users from the Synchronize From Workbench

Add Users – Leave all assigned users for this Workbench alone, then add any new users from Synchronize From Workbench that are not already assigned.

Chapter 8: Gadgets

Chapter Summary

Gadgets are small, self-contained applications that run targeted functions. You may be familiar with Yahoo! Widgets that show current weather, stock quotes, etc., as small, separate windows on the desktop. Xalt | Mobility lets you take a particular Data Object and show it in its own window, called a Gadget. Gadgets use the same functionality and security as the normal rich web application. Gadgets can be launched directly from the desktop without having to first open a Web browser and navigate to a URL.



Rejected Orders

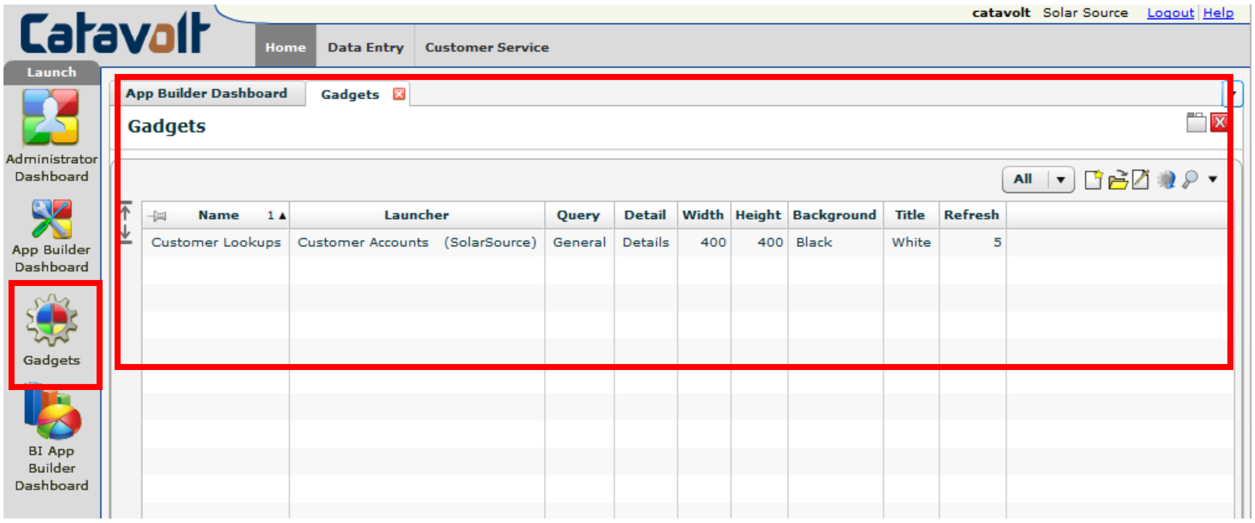
Approver	Order 1 ▼	Amount	Vendor	Buye
josephp	P000676	11	NIPPONDENSO - JAPAN	Steve Wa
josephp	P000675	11	NIPPONDENSO - JAPAN	Steve Wa

Copyright © 2008-2010 Catavolt 5351

Figure 1: Example gadget named Rejected Orders

Accessing Gadgets

Gadgets can be accessed by clicking on the Gadgets Launcher on the Xalt Home Workbench.



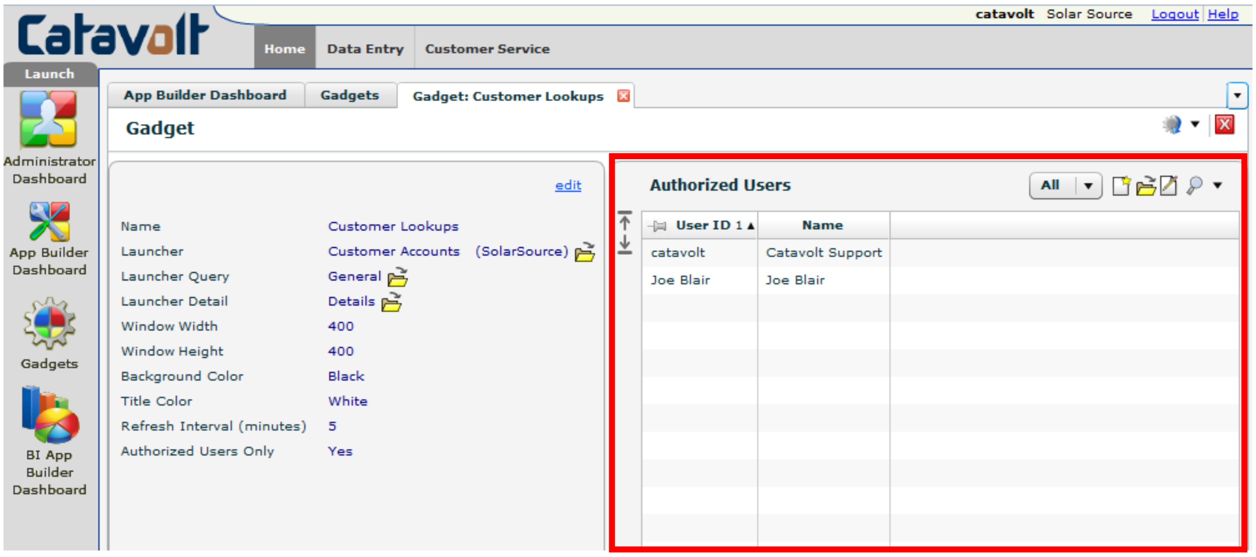
Gadgets

Name 1 ▲	Launcher	Query	Detail	Width	Height	Background	Title	Refresh
Customer Lookups	Customer Accounts (SolarSource)	General	Details	400	400	Black	White	5

Figure 2: The administrator's Home workbench showing the Gadgets launcher with the Gadgets list view open.

Gadget Components

A Gadget has 1 component: Authorized Users (a list of Users authorized to access the Gadget).



Gadget

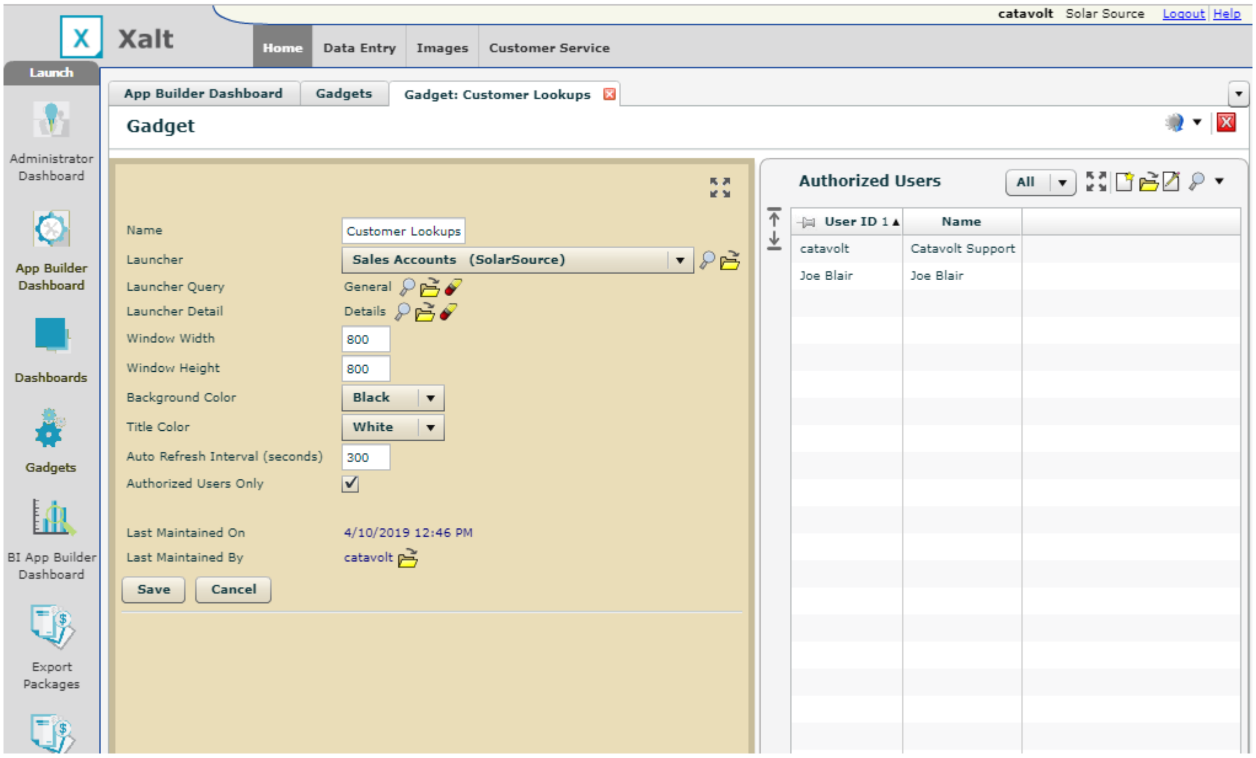
Name	Customer Lookups
Launcher	Customer Accounts (SolarSource)
Launcher Query	General
Launcher Detail	Details
Window Width	400
Window Height	400
Background Color	Black
Title Color	White
Refresh Interval (minutes)	5
Authorized Users Only	Yes

Authorized Users

User ID 1 ▲	Name
catavolt	Catavolt Support
Joe Blair	Joe Blair

Figure 3: The Gadget details view with the Authorized Users query section highlighted.

Creating Gadgets



Name: Customer Lookups
Launcher: Sales Accounts (SolarSource)
Launcher Query: General
Launcher Detail: Details
Window Width: 800
Window Height: 800
Background Color: Black
Title Color: White
Auto Refresh Interval (seconds): 300
Authorized Users Only: ☑

Last Maintained On: 4/10/2019 12:46 PM
Last Maintained By: catavolt

User ID 1 ▲	Name
catavolt	Catavolt Support
Joe Blair	Joe Blair

Figure 4: The create Gadget details view.

When creating a Gadget, you must specify the Name. The Name will be used as the Title of the Gadget Window as well as the name of the Gadget file that will be downloaded.

Launcher specifies the Launcher that is performed when the user runs the Gadget. Launcher displays a list of all Launchers in the system.

Launcher Query specifies the initial Query that will be displayed when the user runs the Gadget. If you leave this value blank, the Data Object's Default Rich Query will be used.

Launcher Detail specifies the initial Detail that will be displayed when the user runs the Gadget and opens a record from a Query. If you leave this value blank, the Data Object's Default Rich Detail will be used.

Window Width specifies the initial width of the Gadget window that is displayed when the user runs the Gadget. Depending on the type of Gadget used, this value may be changed by the end user when they run the Gadget.

Window Height specifies the initial height of the Gadget window that is displayed when the user runs the Gadget. Depending on the type of Gadget used, this value may be changed by the end user when they run the Gadget.

Background Color specifies the border background color of the Gadget window that is displayed when the user runs the Gadget. Depending on the type of Gadget used, this value may be changed by the end user when they run the Gadget.

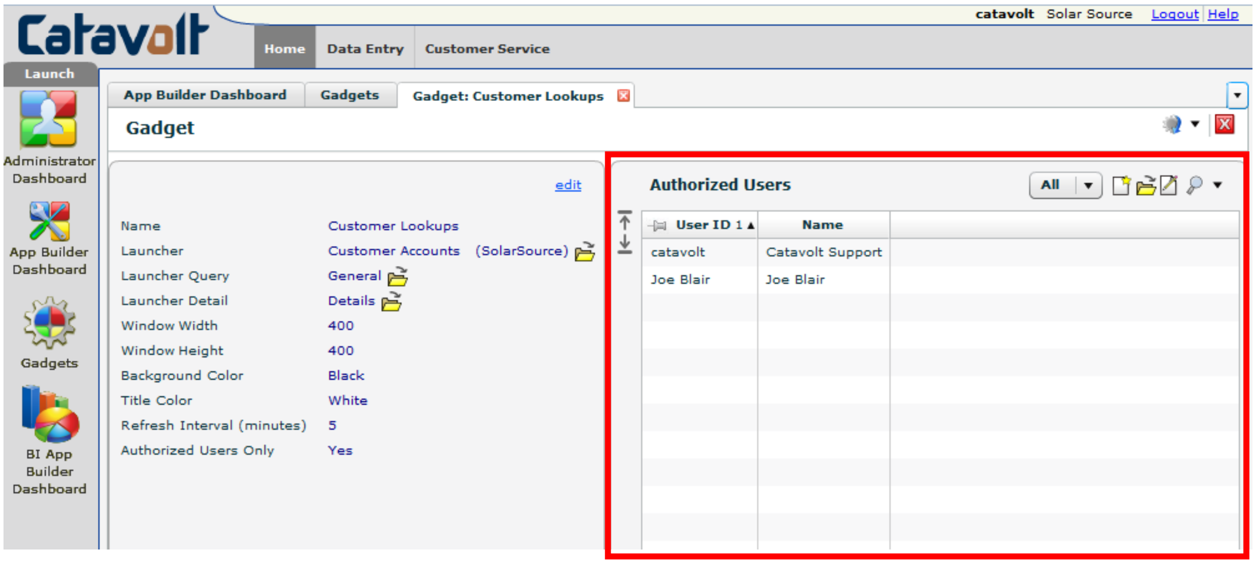
Title Color specifies the border text color of the Gadget window that is displayed when the user runs the Gadget. Depending on the type of Gadget used, this value may be changed by the end user when they run the Gadget.

Auto Refresh Interval (seconds) specifies the amount of time that should pass before the Gadget window automatically refreshes itself. Allowable values are 0 (no auto-refresh) or between 5 and 14400 seconds (4 hours). The default value is 300 (5 minutes). Note that all Gadgets have a menu option to manually refresh the window regardless of the refresh interval setting. Turning on auto-refresh has the potential to greatly increase the load on your back-end server depending on the time set and the number of users accessing this Gadget. When Auto Refresh is applied to a Gadget, the setting is used for the entire Gadget window, including all sub-sections. If the Gadget contains sections that point to other Data Objects that also have auto-refresh enabled, the refresh timers for those Data Objects are ignored. The refresh timers for those Data Object will be applied again when drilling into Details for those Data Objects.

Authorized Users Only specifies whether anyone can run the Gadget or only a specific list of users. If Authorized Users Only is set to Yes, you must enter the list of user profiles in the Authorized Users section. If Authorized Users Only is set to No, any user who receives the Gadget may run it. Note that regardless of the Authorized Users Only Setting, all users must supply a valid User ID/password when running the Gadget.

Authorized Users

Authorized Users specifies the list of User Profiles that are authorized to run the Gadget. If the Authorized Users Only field is set to No, then this list will be empty and all User Profiles will be authorized to run the specified Gadget.

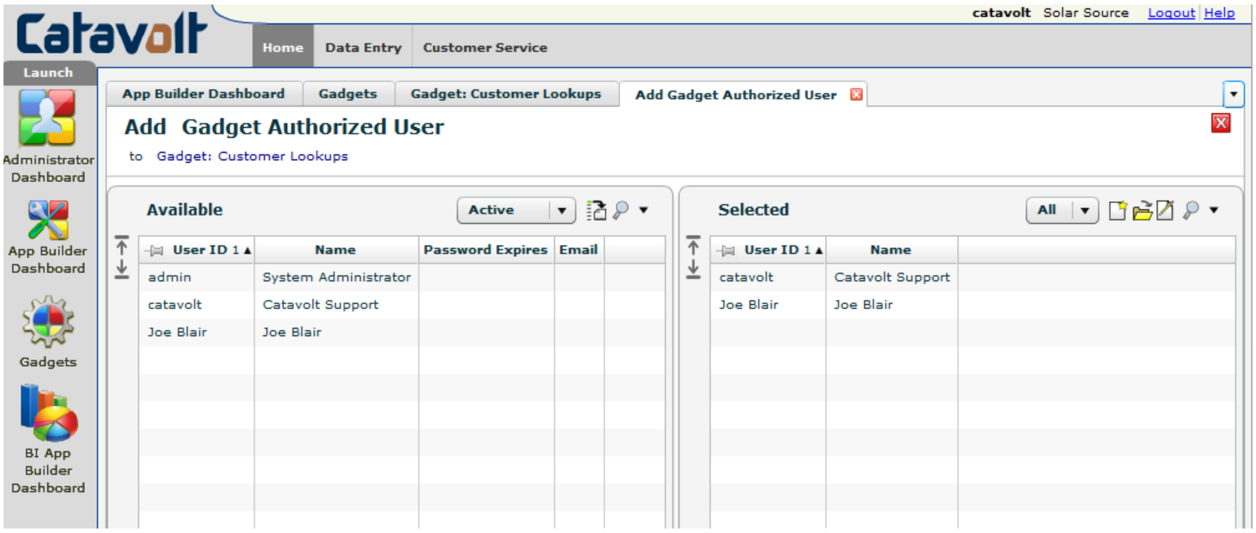


Name	Customer Lookups
Launcher	Customer Accounts (SolarSource)
Launcher Query	General
Launcher Detail	Details
Window Width	400
Window Height	400
Background Color	Black
Title Color	White
Refresh Interval (minutes)	5
Authorized Users Only	Yes

User ID 1 ▲	Name
catavolt	Catavolt Support
Joe Blair	Joe Blair

Figure 5: The Gadget details view with the Authorized Users query section highlighted.

When adding Authorized Users, you will be presented with two lists. The Available User list shows all users. The Selected Users list shows the users that are authorized to access the Gadget.



Add Gadget Authorized User

to Gadget: Customer Lookups

Available

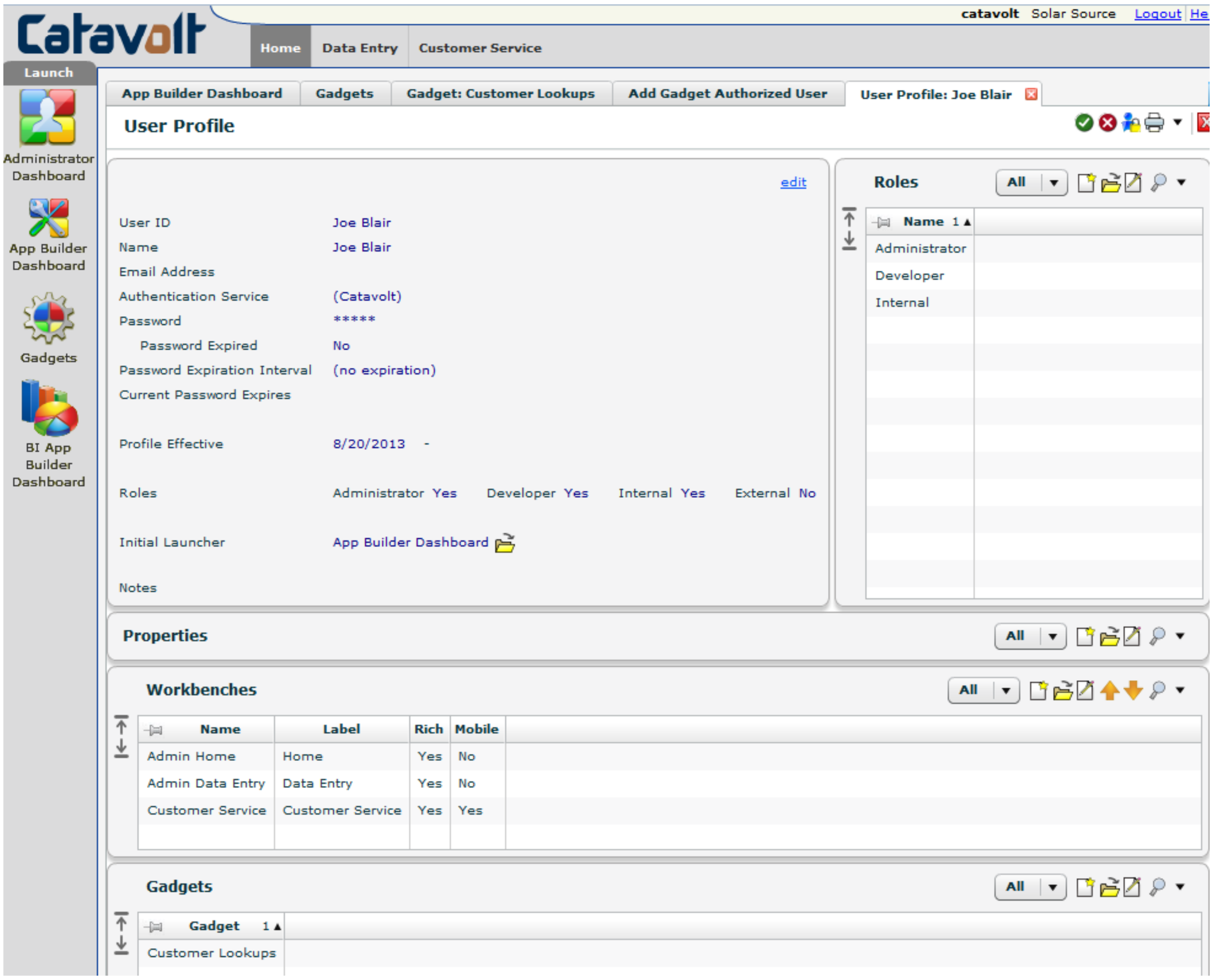
User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Selected

User ID 1 ▲	Name
catavolt	Catavolt Support
Joe Blair	Joe Blair

Figure 6: The Add Gadget Authorized User view.

You can select a single or multiple users and press the Add button to authorize them to the Gadget. After the Gadget Authorized User has been created, opening the Authorized User record will display the details for the User. See Chapter 9: User Profile Administration for more information about User Profiles.



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

Workbenches

Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

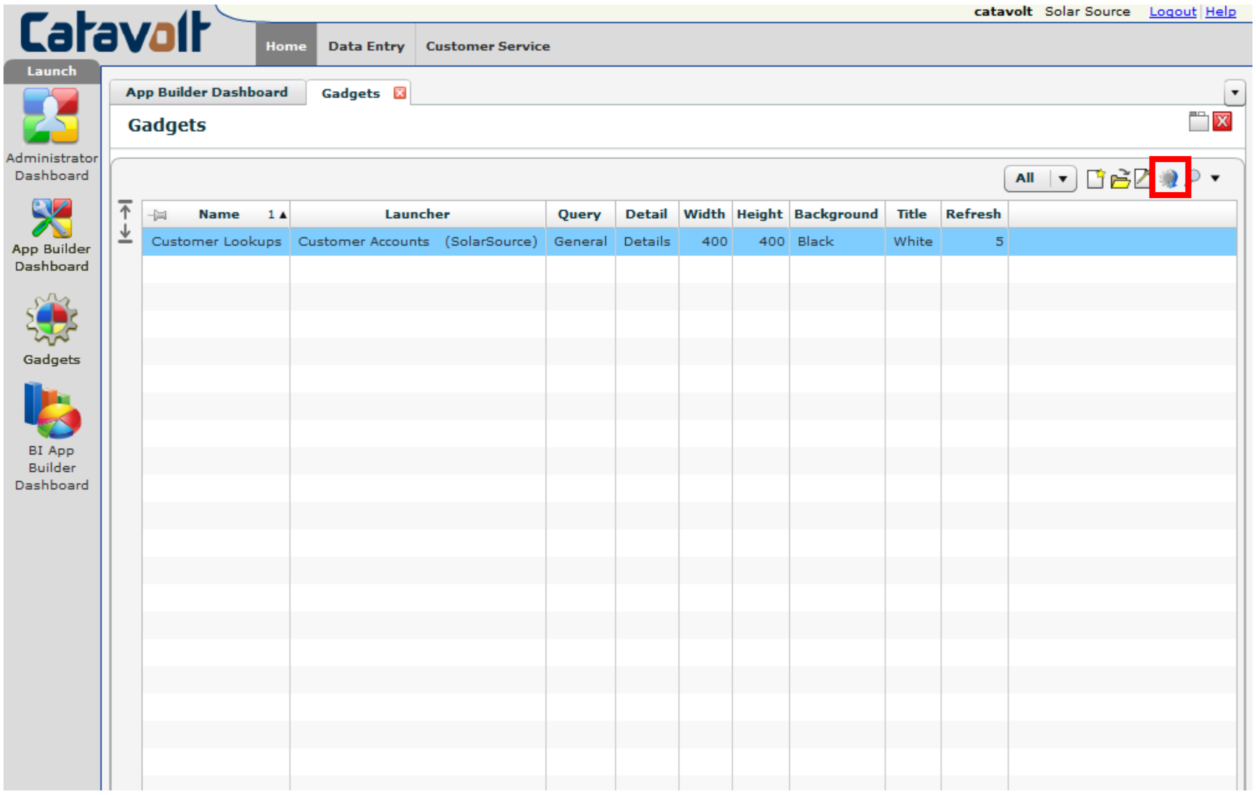
Gadgets

Gadget 1 ▲
Customer Lookups

Figure 7: The User Profile details view.

Deploying Gadgets

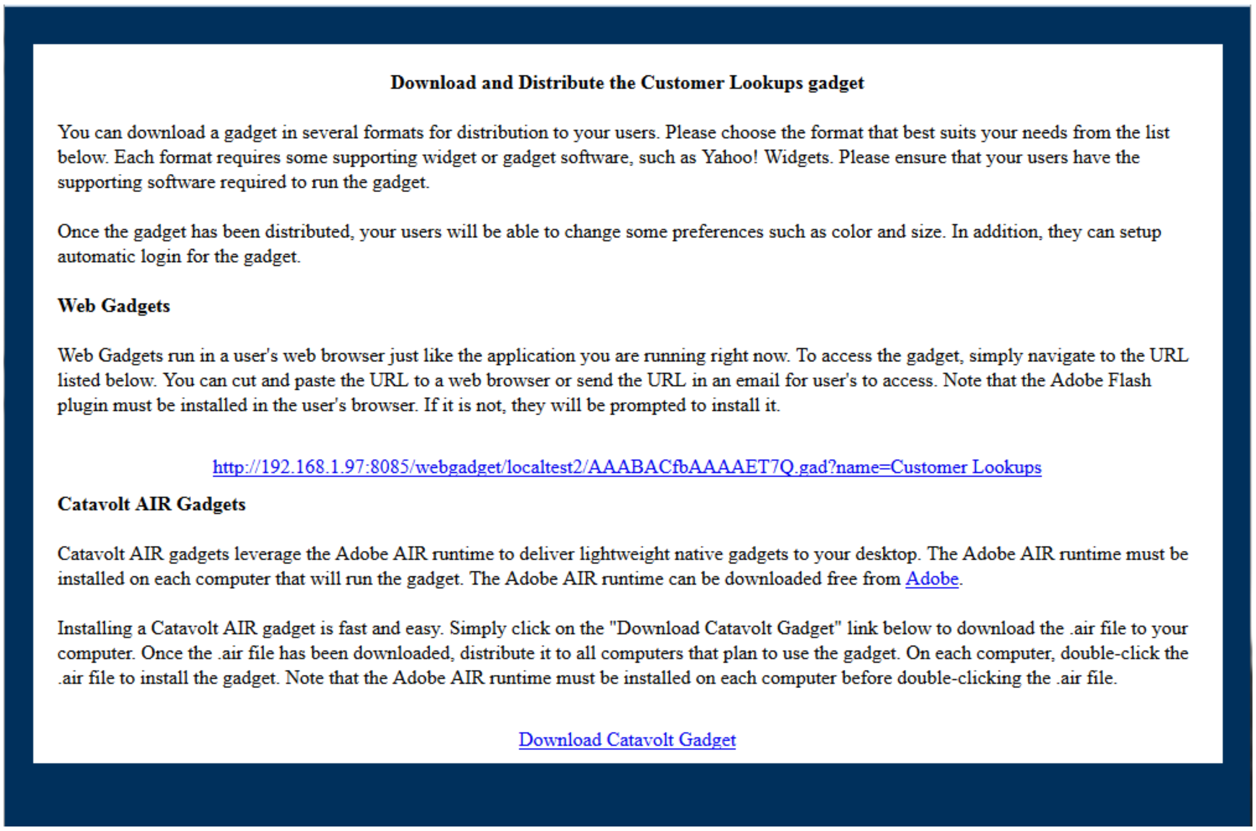
Once a Gadget has been created, it must then be deployed to the people that will be using it. To do this, select a Gadget record and choose the Download Gadget menu option:



Name 1 ▲	Launcher	Query	Detail	Width	Height	Background	Title	Refresh
Customer Lookups	Customer Accounts (SolarSource)	General	Details	400	400	Black	White	5

Figure 8: The Gadget list view with the Download Gadget toolbar button highlighted.

The following page will be displayed:



Download and Distribute the Customer Lookups gadget

You can download a gadget in several formats for distribution to your users. Please choose the format that best suits your needs from the list below. Each format requires some supporting widget or gadget software, such as Yahoo! Widgets. Please ensure that your users have the supporting software required to run the gadget.

Once the gadget has been distributed, your users will be able to change some preferences such as color and size. In addition, they can setup automatic login for the gadget.

Web Gadgets

Web Gadgets run in a user's web browser just like the application you are running right now. To access the gadget, simply navigate to the URL listed below. You can cut and paste the URL to a web browser or send the URL in an email for user's to access. Note that the Adobe Flash plugin must be installed in the user's browser. If it is not, they will be prompted to install it.

http://192.168.1.97:8085/webgadget/localtest2/AAABACfbAAAAET7Q.gad?name=Customer Lookups

Catavolt AIR Gadgets

Catavolt AIR gadgets leverage the Adobe AIR runtime to deliver lightweight native gadgets to your desktop. The Adobe AIR runtime must be installed on each computer that will run the gadget. The Adobe AIR runtime can be downloaded free from Adobe.

Installing a Catavolt AIR gadget is fast and easy. Simply click on the "Download Catavolt Gadget" link below to download the .air file to your computer. Once the .air file has been downloaded, distribute it to all computers that plan to use the gadget. On each computer, double-click the .air file to install the gadget. Note that the Adobe AIR runtime must be installed on each computer before double-clicking the .air file.

Download Catavolt Gadget

Figure 9: The download Gadget web page.

You have 2 options for distributing the Gadget:

- Web Gadget – Runs under Adobe Flash inside a Web page. Simply paste the specified URL into a Web browser.
- Adobe AIR Gadget – Runs under Adobe Air as a separate window on the desktop. Download the .air file and send it to users.

The download page (as shown) contains full instructions on how to deploy the various Gadget types.

Chapter 9: User Profile Administration

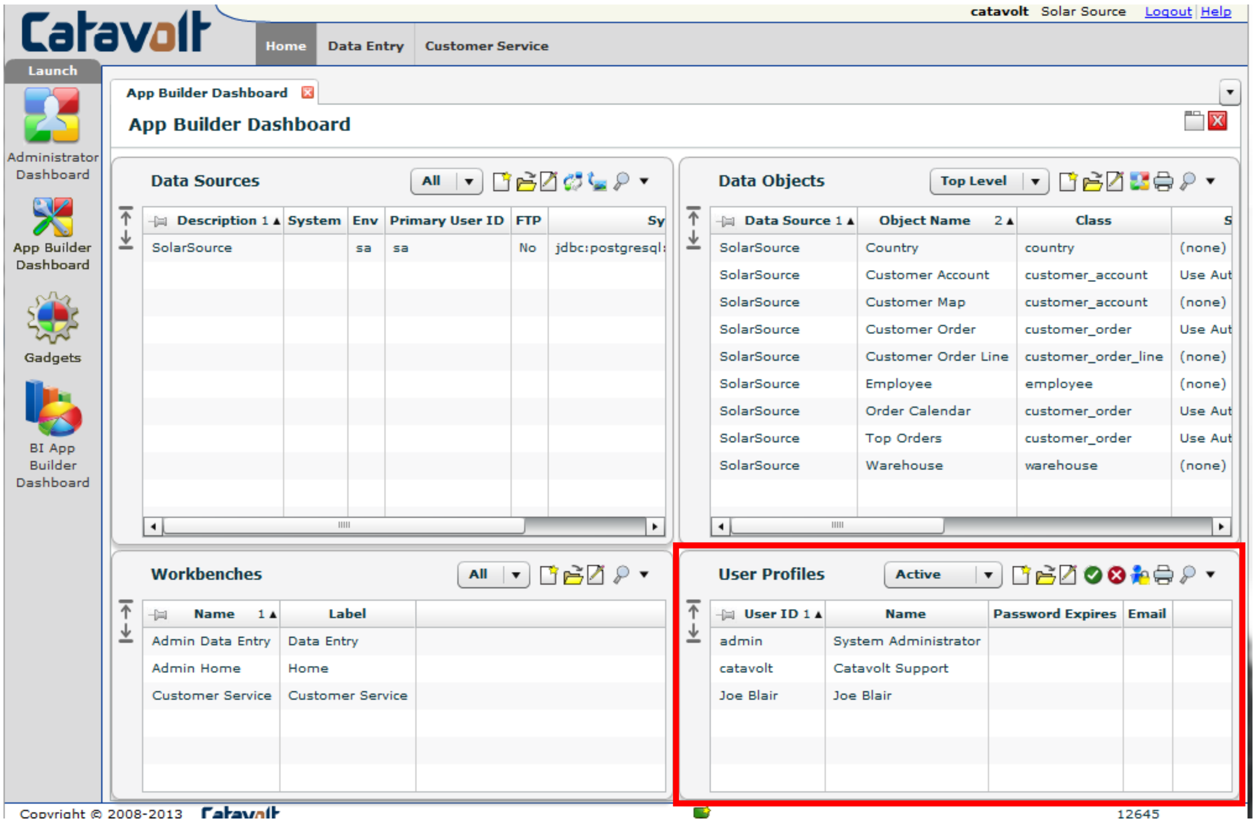
Chapter Summary

A User Profile is created for each Xalt | Mobility user. User Profiles contain the user ID and password that are used to access Xalt | Mobility. The User Profile is also where you define which Workbenches a user is allowed to see and which Gadgets the user is allowed to access.

Some users, usually employees, may already have credentials on the internal system. Other users, such as customers or suppliers, may not. You may define both types of users to Xalt | Mobility. You can set up user profiles so that the internal system can perform authentication for users it recognizes. For all others, you can allow Xalt | Mobility to do the authentication.

Accessing User Profiles

When a developer signs on to Xalt | Mobility, they will typically be immediately brought to the App Builder Dashboard. This dashboard is the hub that allows you to create and deploy Xalt applications. The User Profiles section shows you a list of all User Profiles defined in Xalt sorted by User ID.



Description 1 ▲	System	Env	Primary User ID	FTP	Sy
SolarSource		sa	sa	No	jdbc:postgresql:

Data Source 1 ▲	Object Name 2 ▲	Class	S
SolarSource	Country	country	(none)
SolarSource	Customer Account	customer_account	Use Aut
SolarSource	Customer Map	customer_account	(none)
SolarSource	Customer Order	customer_order	Use Aut
SolarSource	Customer Order Line	customer_order_line	(none)
SolarSource	Employee	employee	(none)
SolarSource	Order Calendar	customer_order	Use Aut
SolarSource	Top Orders	customer_order	Use Aut
SolarSource	Warehouse	warehouse	(none)

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

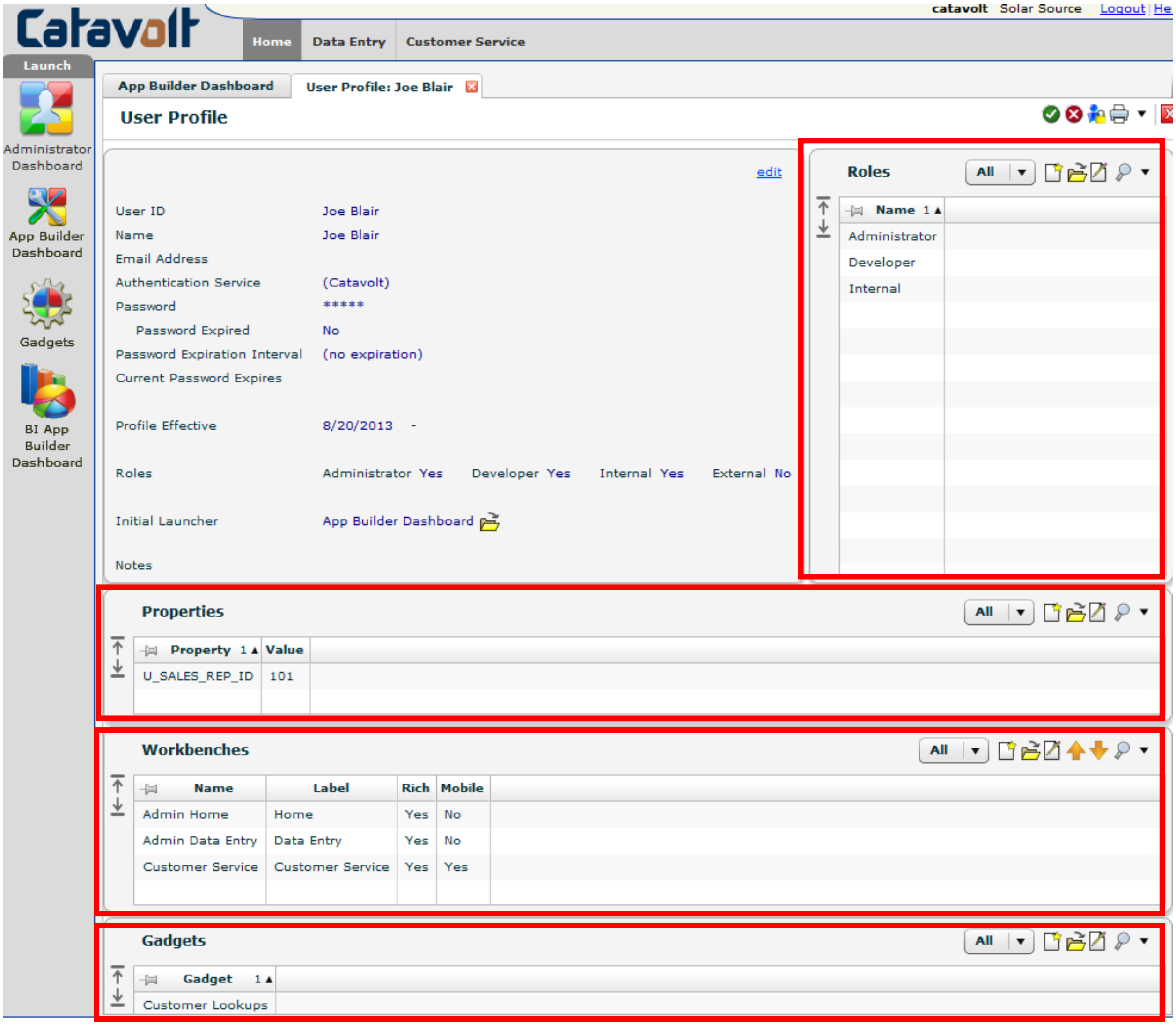
User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Figure 1: The App Builder Dashboard with the User Profiles query section highlighted.

Note that by default, the list of User Profiles only shows Active User Profiles. When User Profiles are suspended, they disappear from the Active list. You can change the dropdown from Active to Inactive or All to see User Profiles that have been suspended. From there, you can reactivate User Profiles that have been previously suspended.

User Profile Components

A User Profile is made up of 4 components: Roles (a list of Security Roles that are assigned to the User Profile), Properties (a list of User Properties for a User Profile), Workbenches (a list of Workbenches that can be accessed by the User Profile) and Gadgets (a list of Gadgets this User Profile is authorized to run).



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

Properties

Property 1 ▲	Value
U_SALES_REP_ID	101

Workbenches

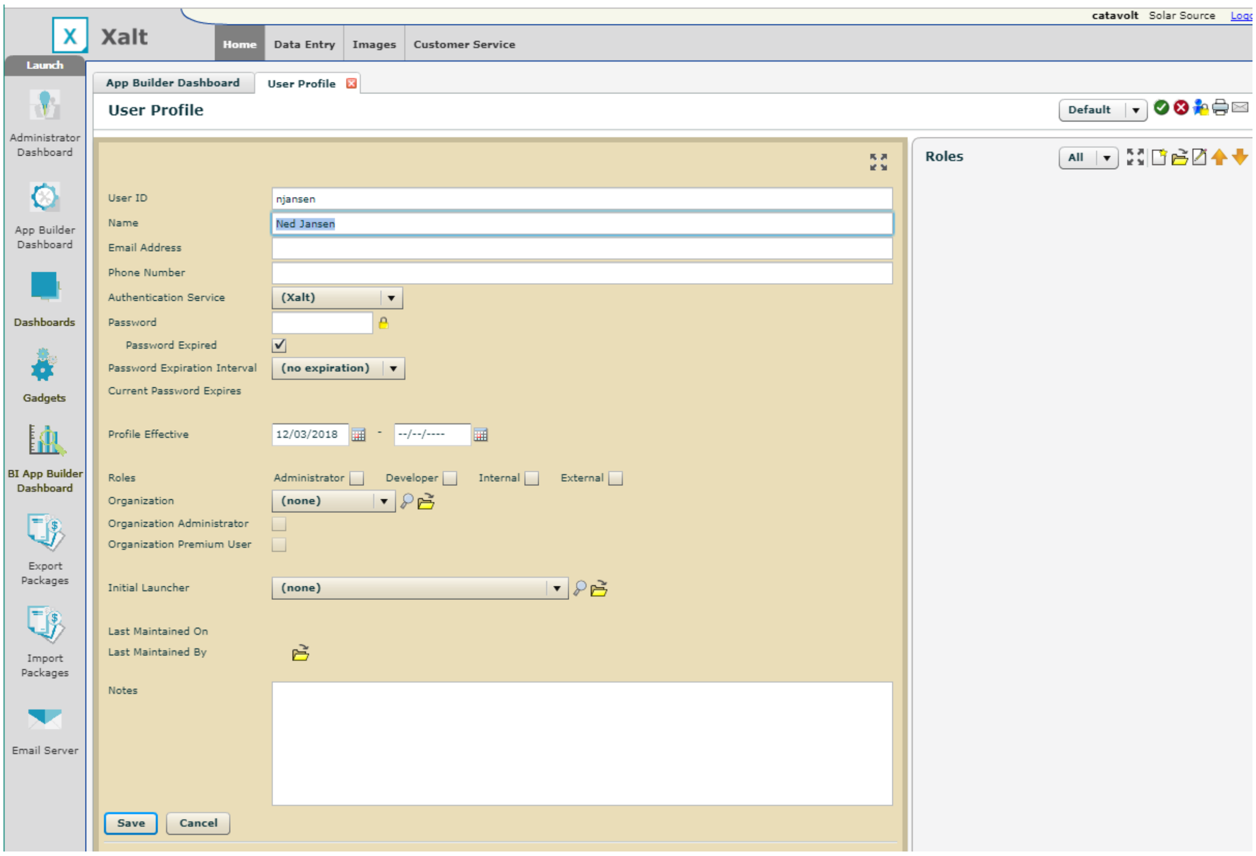
Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

Gadgets

Gadget 1 ▲
Customer Lookups

Figure 2: The User Profile details view with the Properties, Workbenches and Gadgets query sections highlighted.

Creating User Profiles



User Profile

User ID: njansen
Name: Ned Jansen
Email Address:
Phone Number:
Authentication Service: (Xalt)
Password:
Password Expired: ☑
Password Expiration Interval: (no expiration)
Current Password Expires:
Profile Effective: 12/03/2018 - --/--/----
Roles: Administrator □ Developer □ Internal □ External □
Organization: (none)
Organization Administrator: □
Organization Premium User: □
Initial Launcher: (none)
Last Maintained On:
Last Maintained By:
Notes:

Figure 3: The create User Profile details view.

When creating a User Profile, you must specify a User ID. This is the value that must be entered by the user during login.

Name specifies the name of the User Profile. It is used to identify the person using the User ID but is not part of the login process itself.

Email Address specifies the email address of the User Profile.

Phone Number specifies the phone number of the User Profile.

Authentication Service specifies what service will be verifying the user ID and password provided during login by the end user. By default, this value is set to (Xalt), meaning that Hexagon Xalt will verify the user ID and password entered by the end user. Authentication Service also contains a list of all defined Data Sources in Xalt. If you select a Data Source as the authentication source for a User Profile, the user ID and password entered will be sent to the Data Source to verify its validity.

Password specifies the password to be used to verify the user profile. Passwords verified by the Hexagon Xalt Authentication Server must satisfy User Password Requirements specified in Custom Settings. See Chapter 12: Custom Settings for more information about setting password complexity requirements for your tenant. If the Authentication Service is a Data Source, this field is disabled (as the Data Source itself manages the password for the specified User ID).

Password Expired specifies whether the current password should be marked as expired. If this is the case, the next time the user logs into Xalt | Mobility, they will be required to change their password (once it has been verified as the correct current password). If the Authentication Service is a Data Source, this field is disabled (as the Data Source itself manages password expiration for the specified User ID).

Password Expiration Interval (days) specifies how often a user is required to change their Xalt | Mobility password. The default value is blank, meaning the password will never expire. Available values are any integer between 1 and 365 days.

If the Authentication Service is a Data Source, this field is disabled (as the Data Source itself manages password expiration for the specified User ID).

Current Password Expires specifies the date that the current password will expire. This value is automatically calculated based on the Password Expired checkbox and the Password Expiration Interval.

Profile Effective specifies the effective from and to dates that the User Profile is considered active. If a User Profile is not active, it cannot be used to login to Xalt | Mobility even if the password is valid. Since User Profiles cannot be explicitly deleted, making a User Profile no longer effective is how to remove a user from the system. This can be accomplished by changing the effective dates, or by choosing the Suspend menu action for a User Profile.

Roles specifies what activities the user profile can perform in Xalt | Mobility. The following standard roles are available:

- Administrator – Allows the user to create and administer user profiles and other security options
- Developer – Allows the user to create and maintain Xalt data (Data Sources, Data Objects, etc.)
- Internal – Allows the user to connect to Xalt | Mobility as an internal (employee) end user
- External – Allows the user to connect to Xalt | Mobility as an external (customer, supplier, etc.) end user

Other Roles can be created in the Roles section. See Chapter 10: Security for more information about these standard roles.

Organization specifies the name of the Organization this User Profile is assigned to, if any. See the Organizations section in Administrator Dashboard below for more information about Organizations.

Organization Administrator specifies if this User Profile can administer other User Profiles in the same Organization. See the Organizations section in Administrator Dashboard below for more information about Organizations.

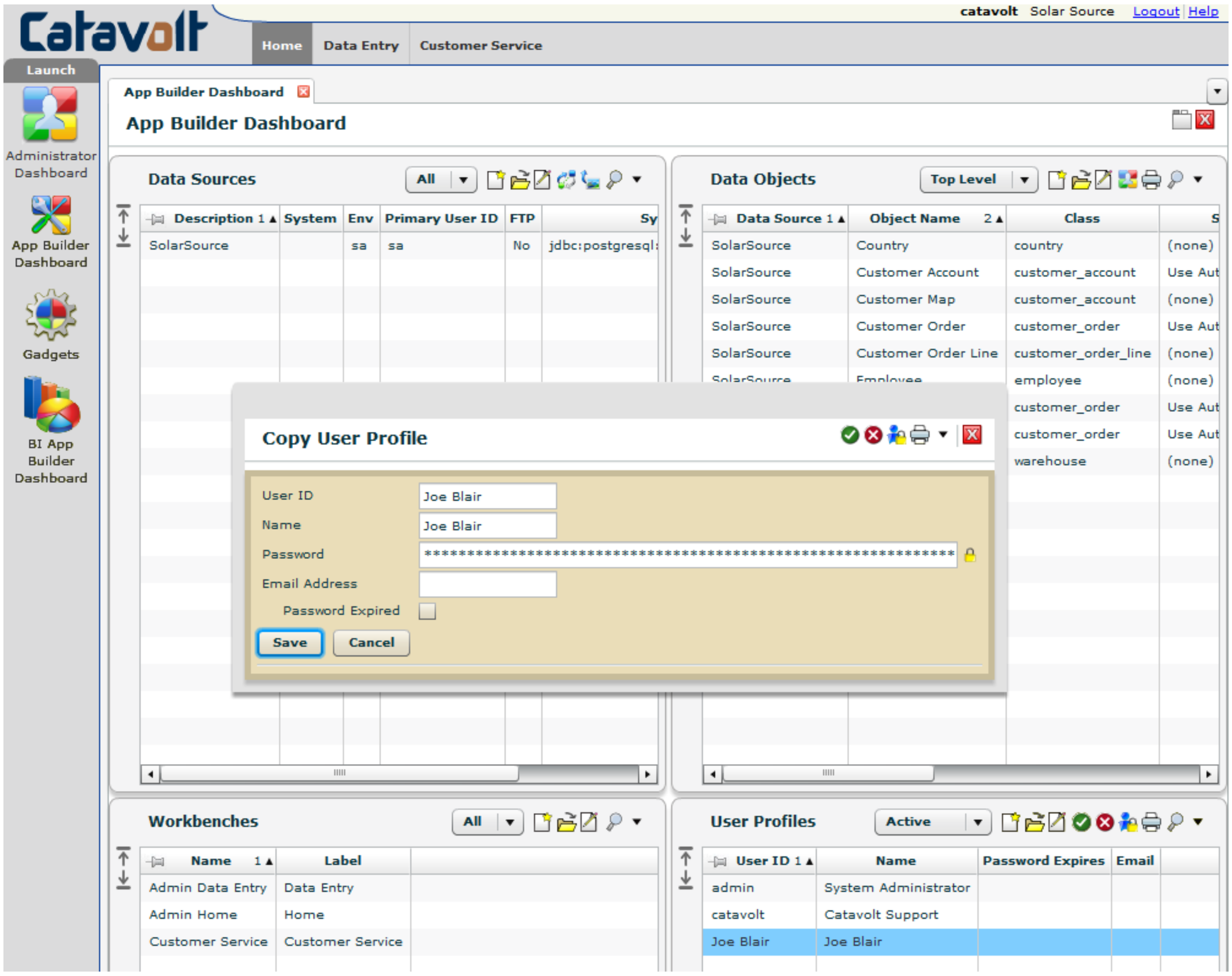
Organization Premium User specifies if this User Profile is a Premium User in their Organization. See the Organizations section in Administrator Dashboard below for more information about Organizations.

Initial Launcher allows you to specify a Launcher to automatically run when the user first logs into Xalt | Mobility. By default, this value is set to (none). This can be useful for users who typically access the same launcher each time they login to Xalt | Mobility. For example, an Xalt developer will typically choose the App Builder Dashboard as their initial launcher.

Notes provides an area to record free-form notes about the user profile (why it was suspended, etc.).

Copying User Profiles

You may have instances where you need to make a copy of an existing User Profile. You can select the Copy menu option to accomplish this. When Copying a User Profile, you will be prompted to supply the new User ID, Name, Password, Email Address, and Email Address. In addition, you can set whether or not the Password entered will be immediately expired. An exact copy of this User Profile along with all of its components will be made.



App Builder Dashboard

Data Sources — All

Description 1 ▲	System	Env	Primary User ID	FTP	Sy
SolarSource		sa	sa	No	jdbc:postgresql:

Data Objects — Top Level

Data Source 1 ▲	Object Name 2 ▲	Class	S
SolarSource	Country	country	(none)
SolarSource	Customer Account	customer_account	Use Aut
SolarSource	Customer Map	customer_account	(none)
SolarSource	Customer Order	customer_order	Use Aut
SolarSource	Customer Order Line	customer_order_line	(none)
SolarSource	Employee	employee	(none)
		customer_order	Use Aut
		customer_order	Use Aut
		warehouse	(none)

Copy User Profile

User ID	Joe Blair
Name	Joe Blair
Password	**
Email Address	
Password Expired	□

Workbenches — All

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

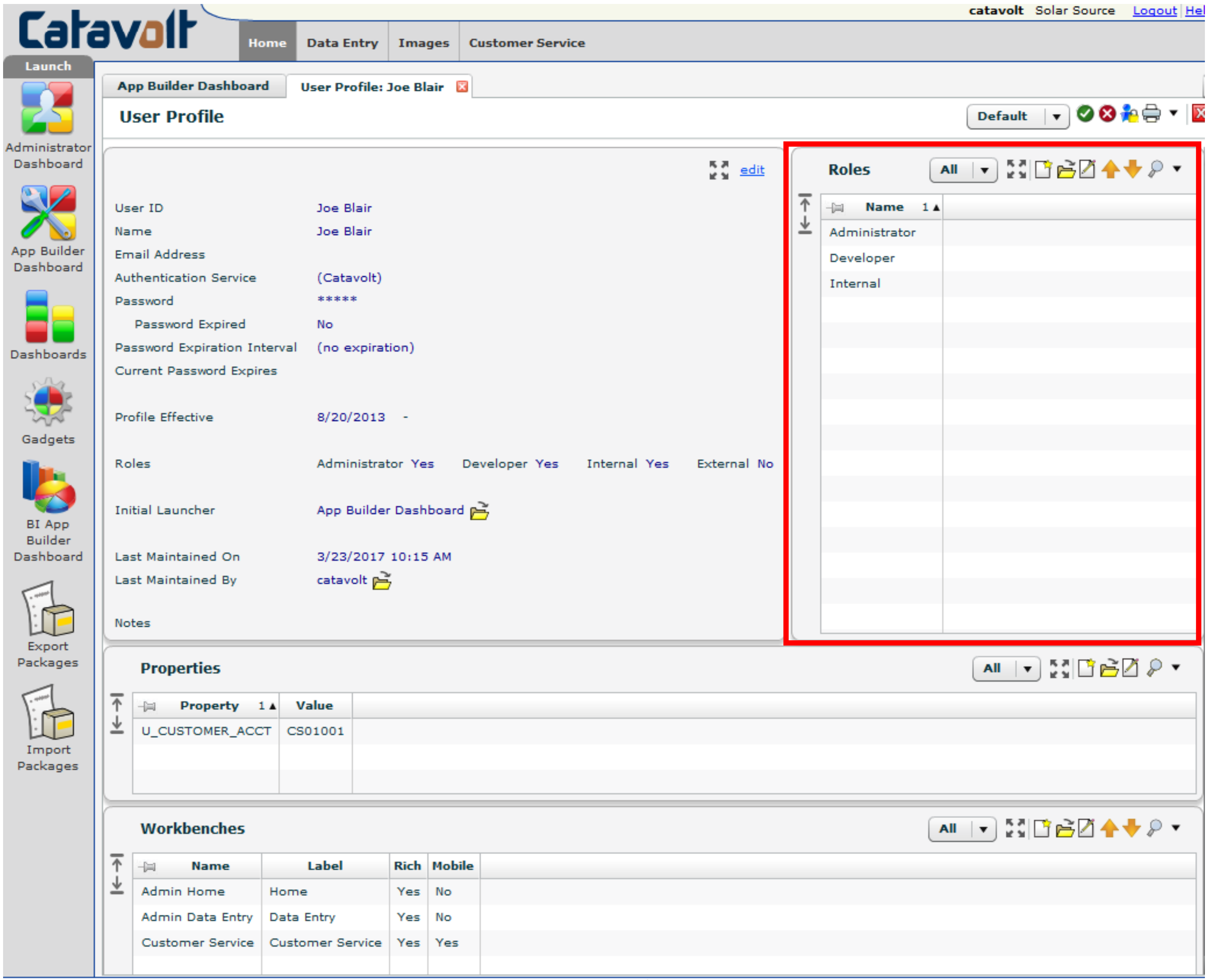
User Profiles — Active

User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Figure 4: The copy User Profile prompt for Data Source Authentication Service

User Profile Roles

User Profile Roles specifies the Security Roles that this User Profile is assigned to. These roles define the object and actions that the user is authorized to access. The 4 Standard Roles (Administrator, Developer, Internal, External) are controlled by the Roles checkboxes but will also appear in the Roles list. In addition, you may create other Security Roles that can be assigned to User Profiles.



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Last Maintained On	3/23/2017 10:15 AM
Last Maintained By	catavolt
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

Properties

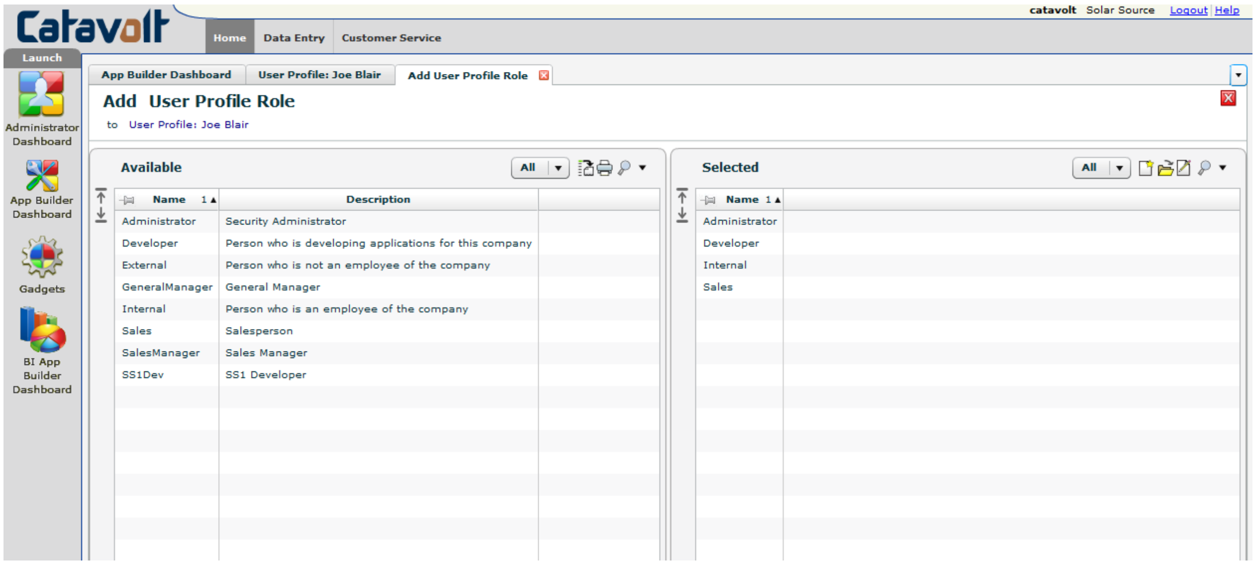
Property 1 ▲	Value
U_CUSTOMER_ACCT	CS01001

Workbenches

Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

Figure 5: The User Profile details view with the Roles query section highlighted.

When adding User Security Roles, you will be presented with two lists. The Available Roles list shows all Security Roles. The Selected Roles list shows the roles that will be assigned to the User Profile.



Add User Profile Role

to User Profile: Joe Blair

Available

Name	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	SS1 Developer

Selected

Name
Administrator
Developer
Internal
Sales

Figure 6: The add User Profile Role view.

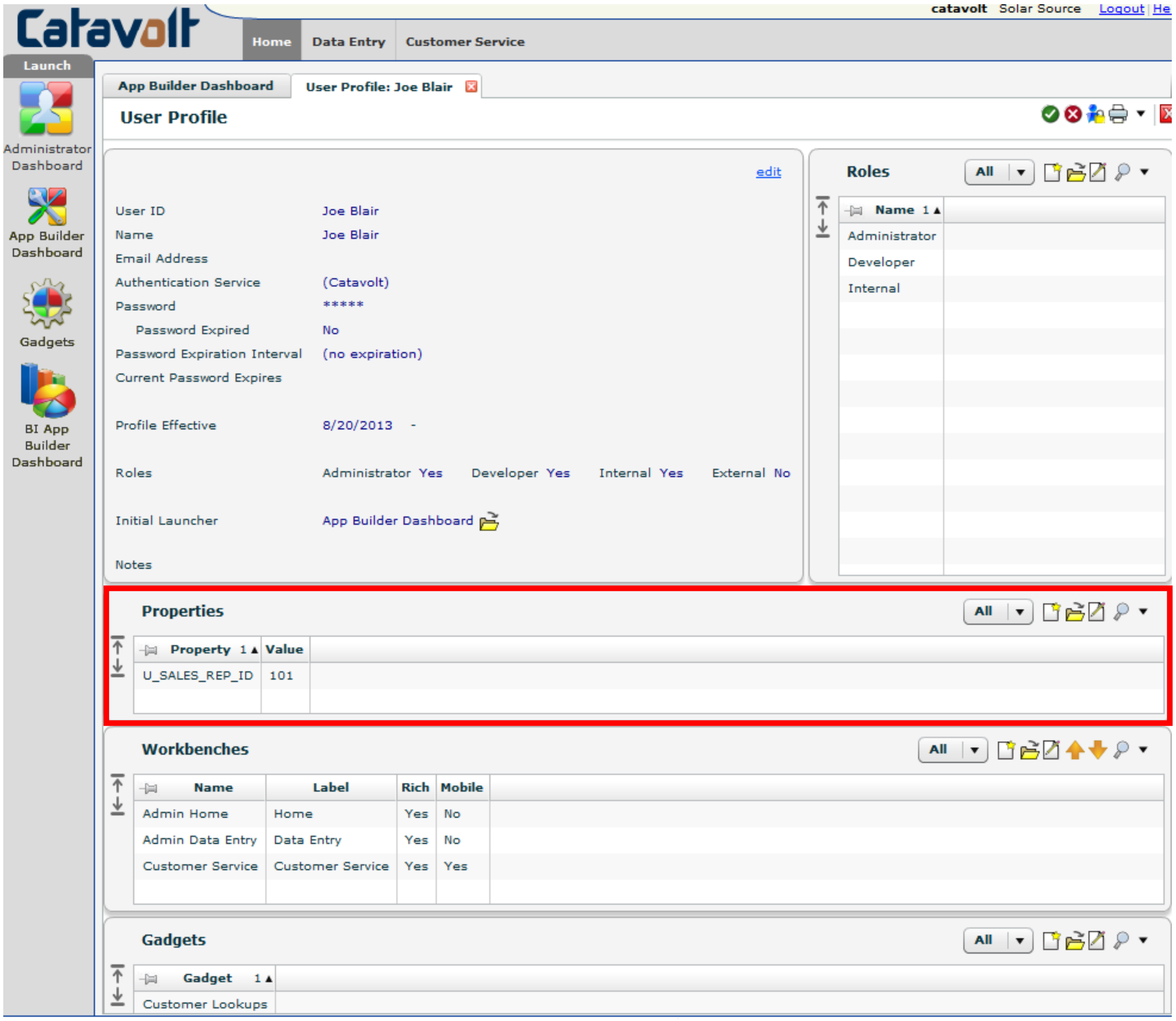
You can select a single or multiple Security Roles and press the Add button to add them to the User Profile.

After the User Profile Role has been created, opening the selected record will display the details for the Security Role.

The Security Roles can be arranged by running the Move Up / Move Down toolbar buttons in order to affect the order of Workbenches assigned to the user via Security Roles. See Chapter 10: Security for more information about Security Roles.

User Profile Properties

User Profile Properties exist for a specific User Profile. These values can be used any place where substitution values are allowed. User Profile Properties are used in places where the data returned or actions performed need to be different depending on the current user. Typical uses are in where clauses to restrict access to certain records or in detail launchers to select a record specific to the current user. If a User Property is used as a substitution value (e.g. on a where clause or Action XML request) and the currently logged in user does not have that property defined, an error will occur when performing the request. If a user is to access a Data Object that contains a User Profile Substitution Value, you must ensure that a corresponding User Profile Property or Default Property exists for that user profile. As these values are case-sensitive, it is typically a best practice to use all capitals when defining these names so as not to conflict with any Data Object property names. See Appendix A: Specifying Messages and Substitution Values for more information about using Substitution Values.



User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

Properties

Property 1 ▲	Value
U_SALES_REP_ID	101

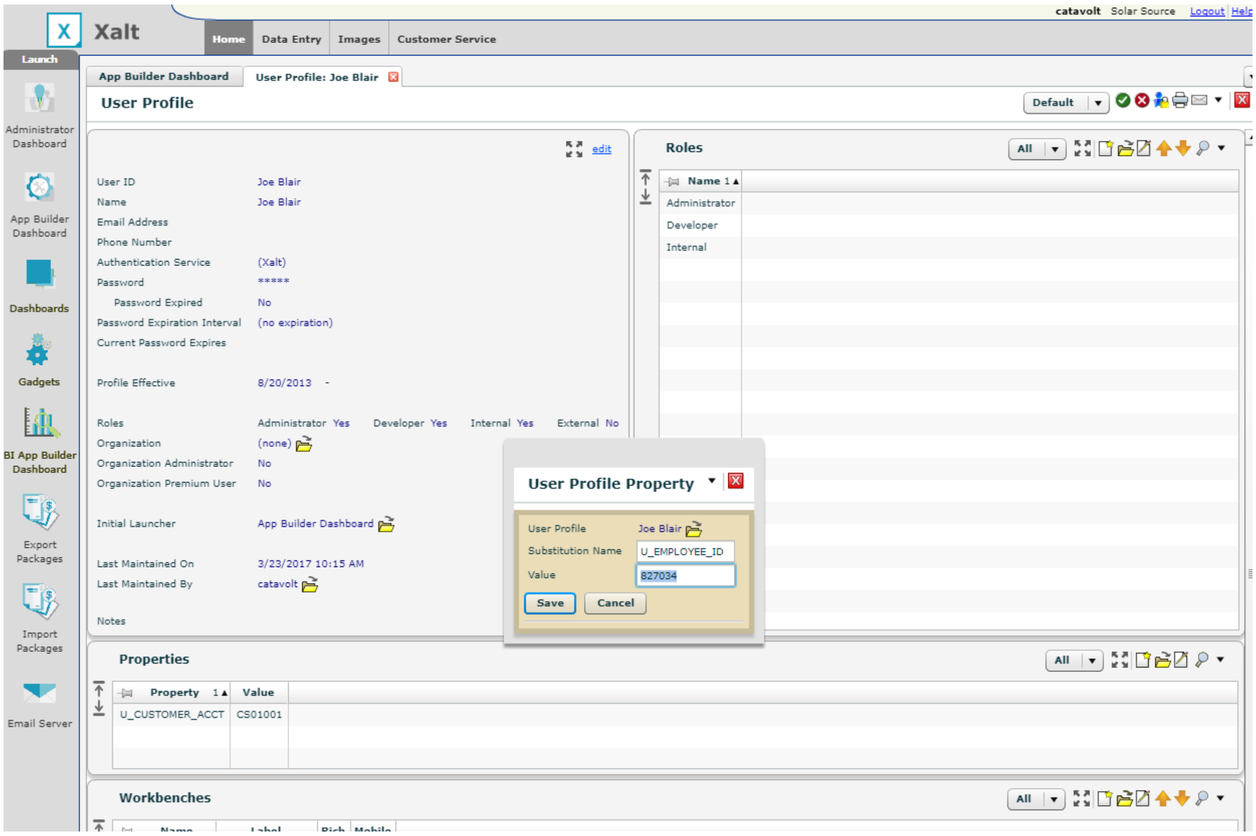
Workbenches

Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

Gadgets

Gadget 1 ▲
Customer Lookups

Figure 7: The User Profile details view with the Properties query section highlighted.



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Phone Number	
Authentication Service	(Xalt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Organization	(none)
Organization Administrator	No
Organization Premium User	No
Initial Launcher	App Builder Dashboard
Last Maintained On	3/23/2017 10:15 AM
Last Maintained By	catavolt
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

User Profile Property

User Profile	Joe Blair
Substitution Name	U_EMPLOYEE_ID
Value	827034

Properties

Property 1 ▲	Value
U_CUSTOMER_ACCT	CS01001

Workbenches

Name	Label	Rich	Mobile

Figure 8: The add User Profile Property prompt.

When creating a User Profile Property, you must specify a Substitution Name. This is the name that will be used to reference the User Profile Property. For instance, you may see something like the following in a Where Clause:

empno = '${U_EMPLOYEE_ID}'

Value specifies the value to be substituted when required. In the example above, the final Where Clause that will be passed to the back end system will be

empno = '51'

Note that the value will be substituted exactly. Using this knowledge, you can get creative with your values. For example, if you wanted to specify a list of employee numbers, you could have a Where Clause of:

empno IN (${U_EMPLOYEE_LIST})

and a U_EMPLOYEE_LIST User Profile Property with a Value of

'50','51','52','99'

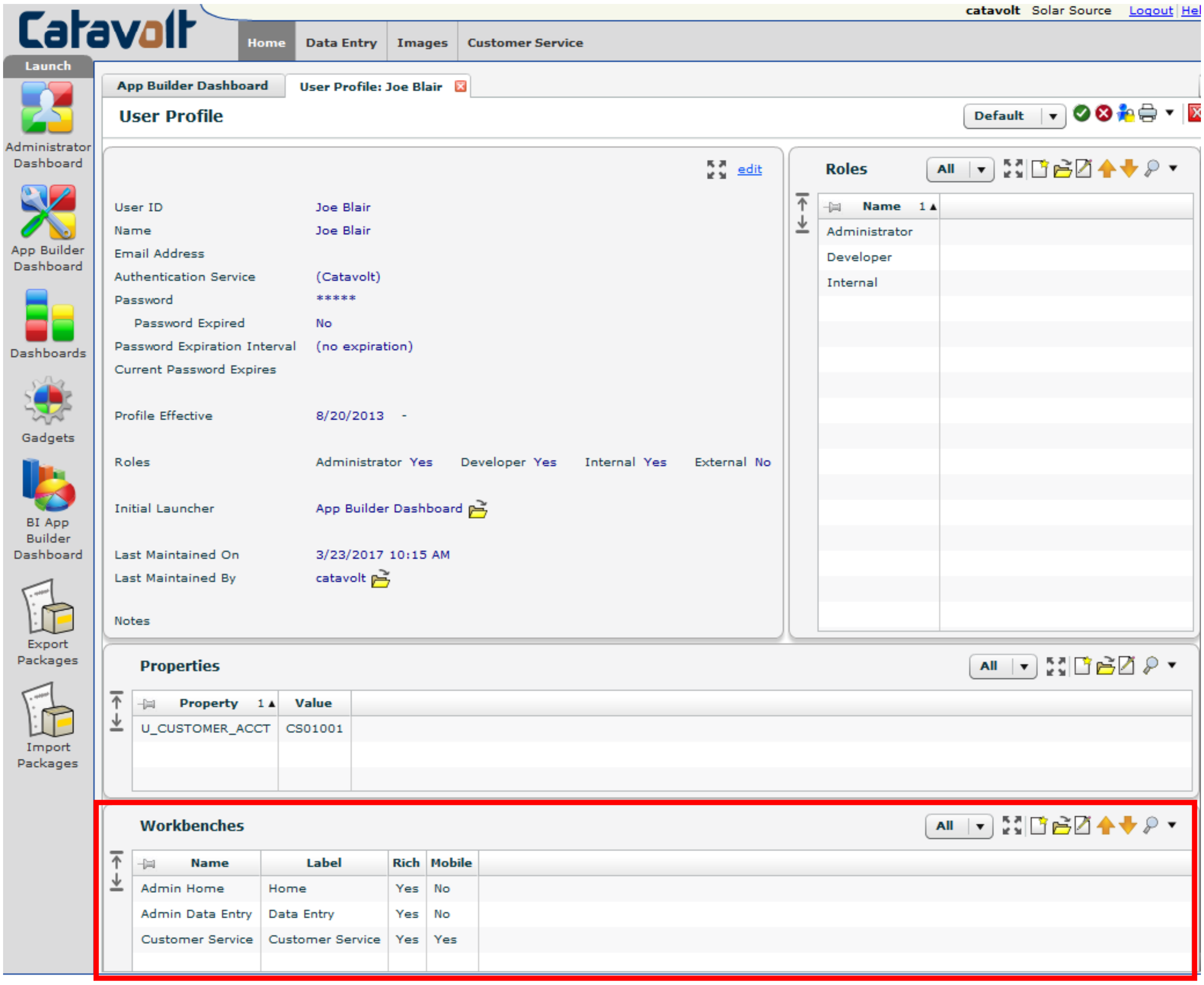
The final Where Clause that will be passed to the back end system will be:

empno IN ('50','51','52','99')

See Appendix A: Specifying Messages and Substitution Values for more information about using Substitution Values.

User Workbenches

User Workbenches specifies which Workbenches can be accessed by the User Profile and in what order they should appear. Multiple Workbenches can be created for a User Profile. They will appear along the Workbench bar in the same order they appear below. To change the order, select a Workbench and press the Move Up and Move Down toolbar buttons.



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Last Maintained On	3/23/2017 10:15 AM
Last Maintained By	catavolt
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

Properties

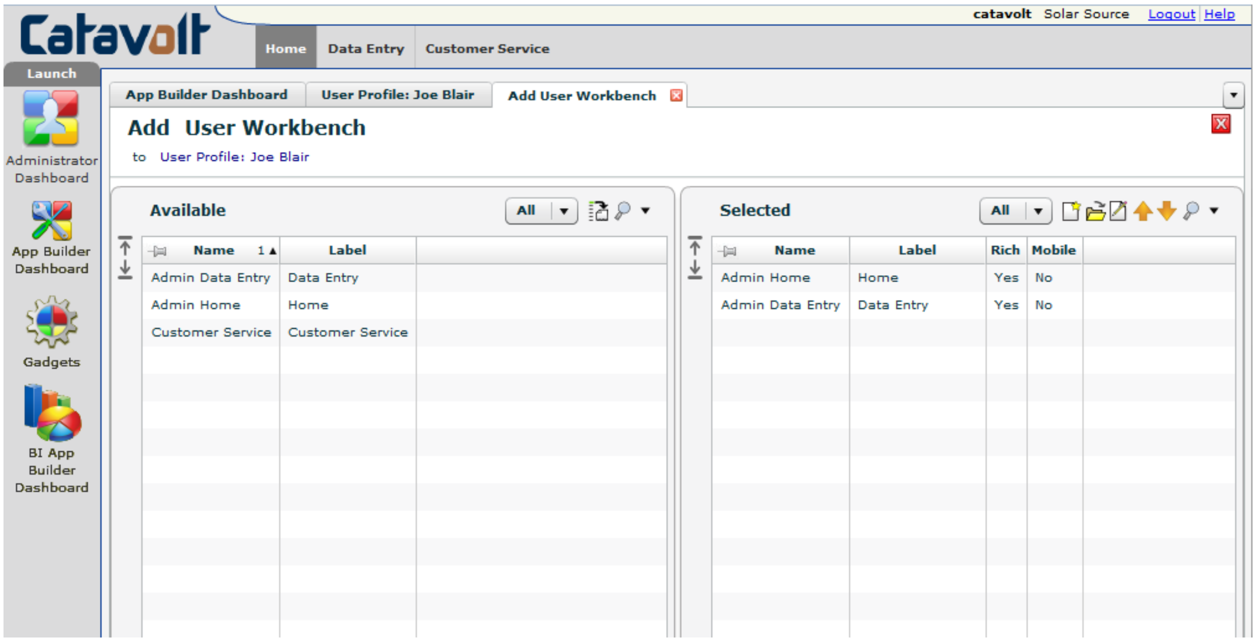
Property 1 ▲	Value
U_CUSTOMER_ACCT	CS01001

Workbenches

Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

Figure 9: The User Profile details view with the Workbenches query list section highlighted.

When adding User Workbenches, you will be presented with two lists. The Available workbenches list shows all Workbenches. The Selected workbenches list shows the workbenches that the user will be running.



Add User Workbench

to User Profile: Joe Blair

Available

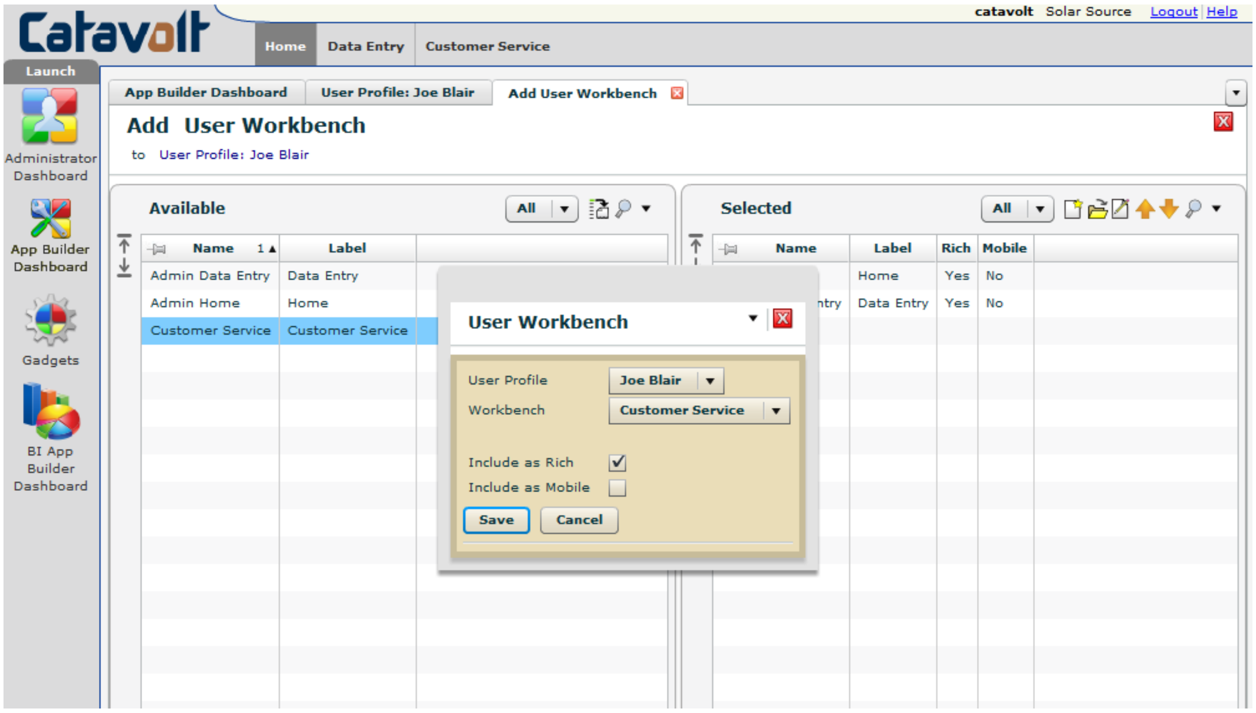
Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

Selected

Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No

Figure 10: The add User Workbench view.

You can select a single or multiple workbenches and press the Add button to add them to the User Profile. If you select a single workbench, you will be presented with the following dialog:



Add User Workbench

to User Profile: Joe Blair

Available

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

Selected

Name	Label	Rich	Mobile
	Home	Yes	No
ntry	Data Entry	Yes	No

User Workbench

User Profile: Joe Blair

Workbench: Customer Service

Include as Rich ☑

Include as Mobile □

Save   Cancel

Figure 11: The add User Workbench view.

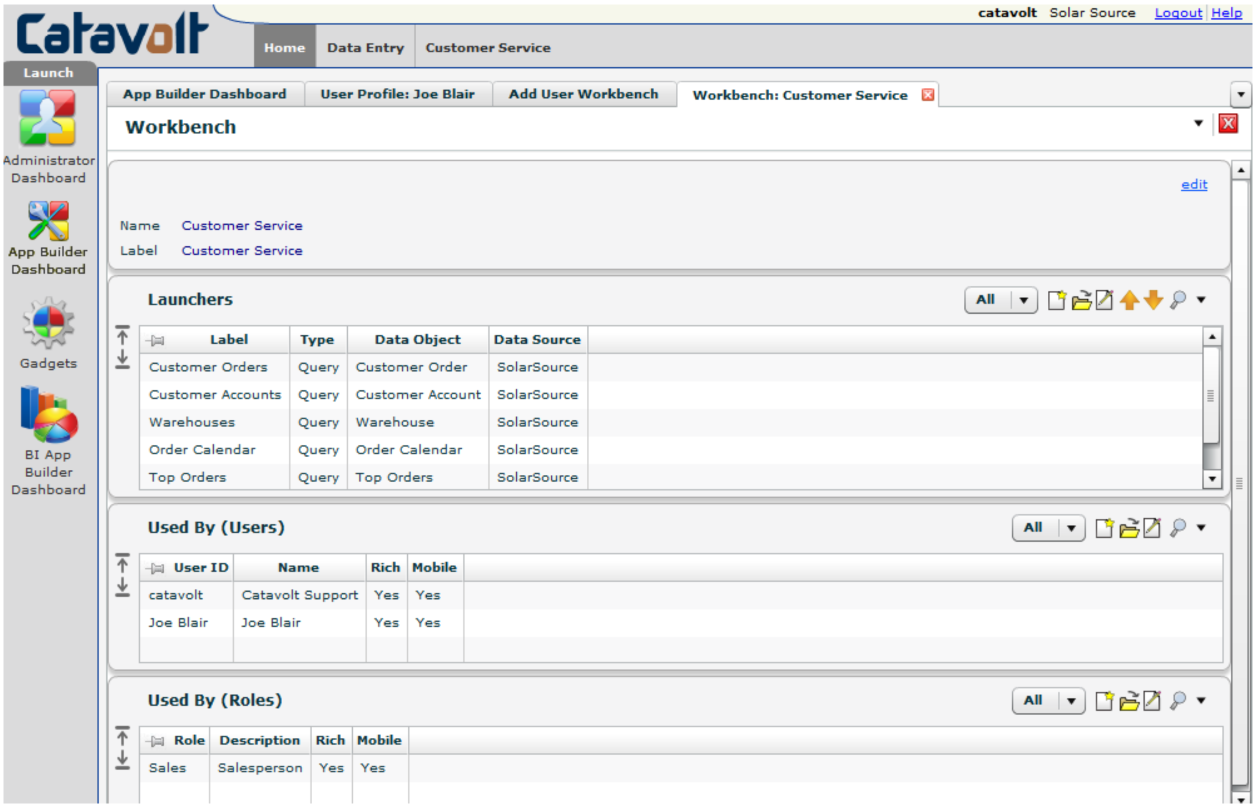
When creating a User Workbench, User Profile specifies the user that is to be granted access to the Workbench. User Profile displays a list of all users in the system. The current User Profile you are working with will be the default selected value.

Workbench specifies the Workbench to be added to the User Profile. Workbench displays a list of all Workbenches in the system.

Include as Rich specifies whether the user should have access to this workbench when running the rich (web) client.

Include as Mobile specifies whether the user should have access to this workbench when running the mobile client.

After the User Workbench has been created, opening the Workbench record will display the details for the Workbench. See Chapter 7: Workbenches for more information about Workbenches.



Workbench

Name Customer Service
Label Customer Service

Launchers

Label	Type	Data Object	Data Source
Customer Orders	Query	Customer Order	SolarSource
Customer Accounts	Query	Customer Account	SolarSource
Warehouses	Query	Warehouse	SolarSource
Order Calendar	Query	Order Calendar	SolarSource
Top Orders	Query	Top Orders	SolarSource

Used By (Users)

User ID	Name	Rich	Mobile
catavolt	Catavolt Support	Yes	Yes
Joe Blair	Joe Blair	Yes	Yes

Used By (Roles)

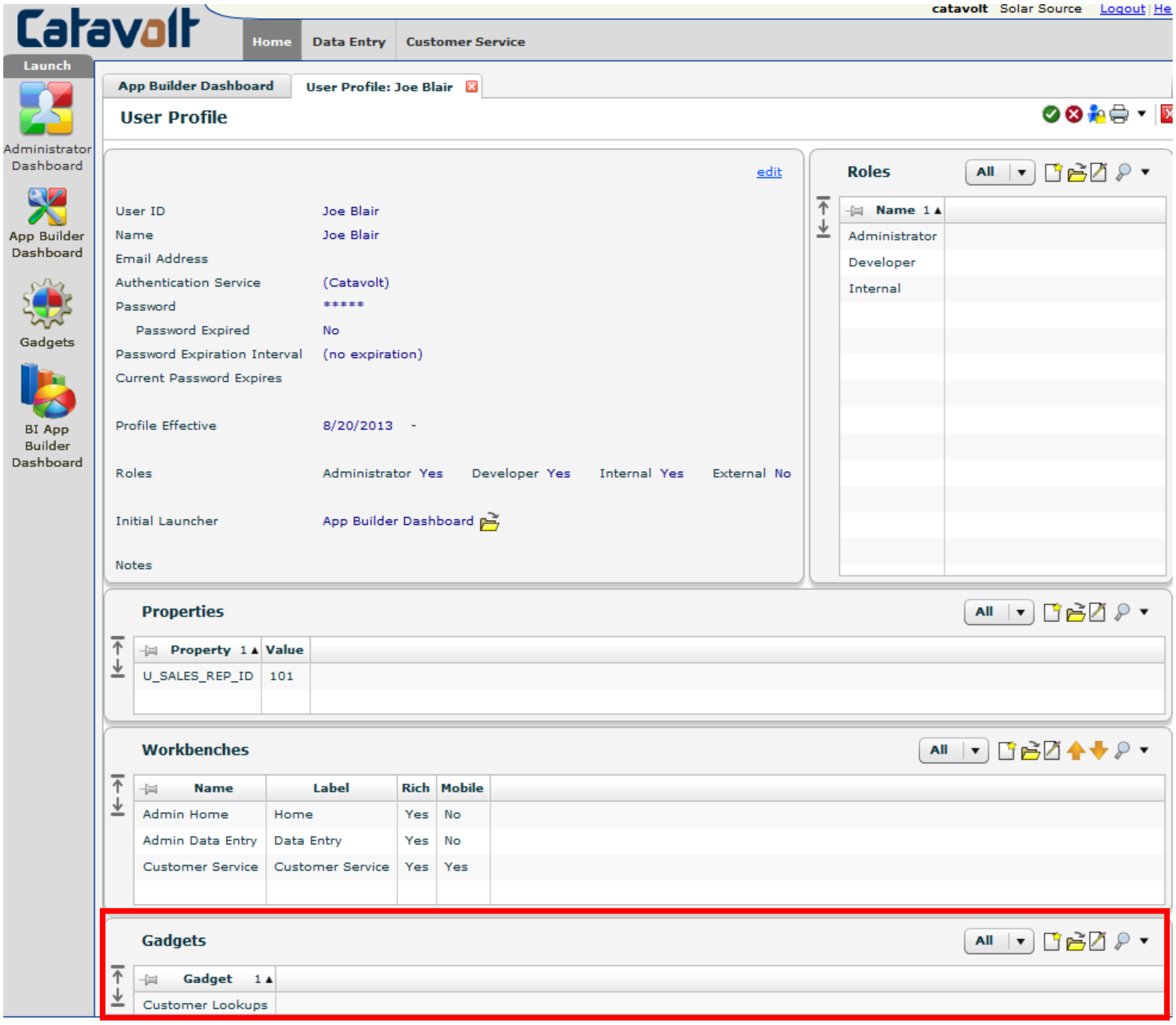
Role	Description	Rich	Mobile
Sales	Salesperson	Yes	Yes

Figure 12: The Workbench details view.

User Profiles that do not have Workbenches defined on the profile itself will automatically pick up Workbenches from the Security Roles that they are assigned to. You can alter the order the Workbenches appear by arranging the list of Security Roles on the User Profile as well as arranging the list of Workbenches on the Security Profile. Note that if a Workbench exists on multiple Security Roles, the first instance will be used (duplicate appearances will be removed).

Authorized Gadgets

Gadgets specifies which Gadgets a User Profile is specifically authorized to access. A User Profile can be authorized to access multiple Gadgets. Note that Gadgets that have set Authorized Users Only to No will not appear in the list even though the User Profile will be able to access them.



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

Properties

Property 1 ▲	Value
U_SALES_REP_ID	101

Workbenches

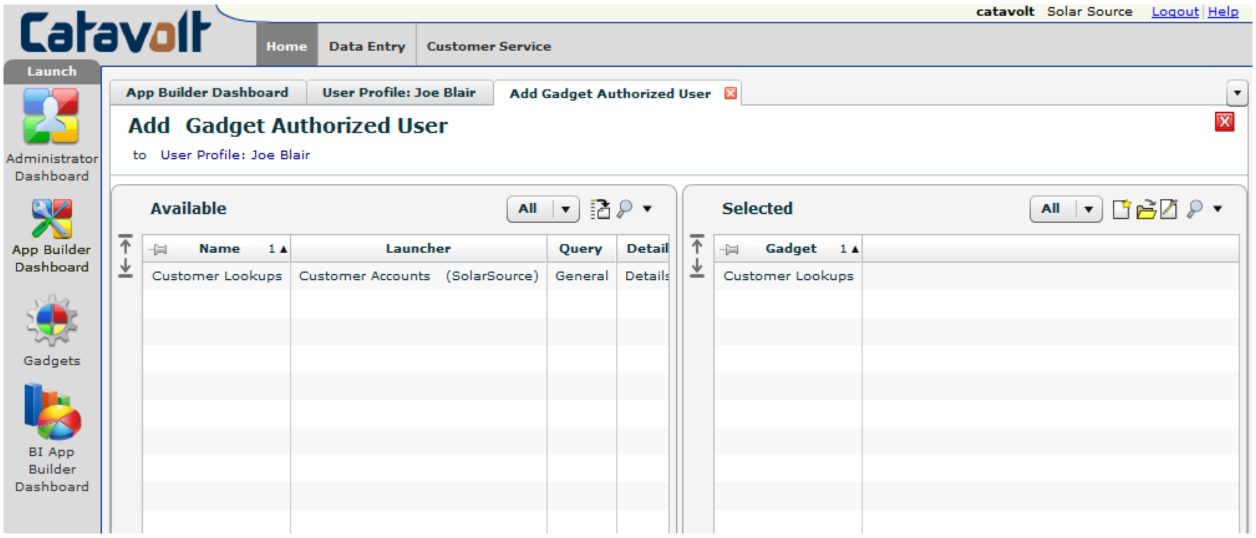
Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

Gadgets

Gadget 1 ▲
Customer Lookups

Figure 13: The User Profile details view with the Gadgets query section highlighted.

When adding Gadget Authorized Users, you will be presented with two lists. The Available Gadgets list shows all Gadgets. The Selected Gadgets list shows the gadgets that the user is authorized to run.



Add Gadget Authorized User

to User Profile: Joe Blair

Available

Name 1 ▲	Launcher	Query	Detail
Customer Lookups	Customer Accounts (SolarSource)	General	Details

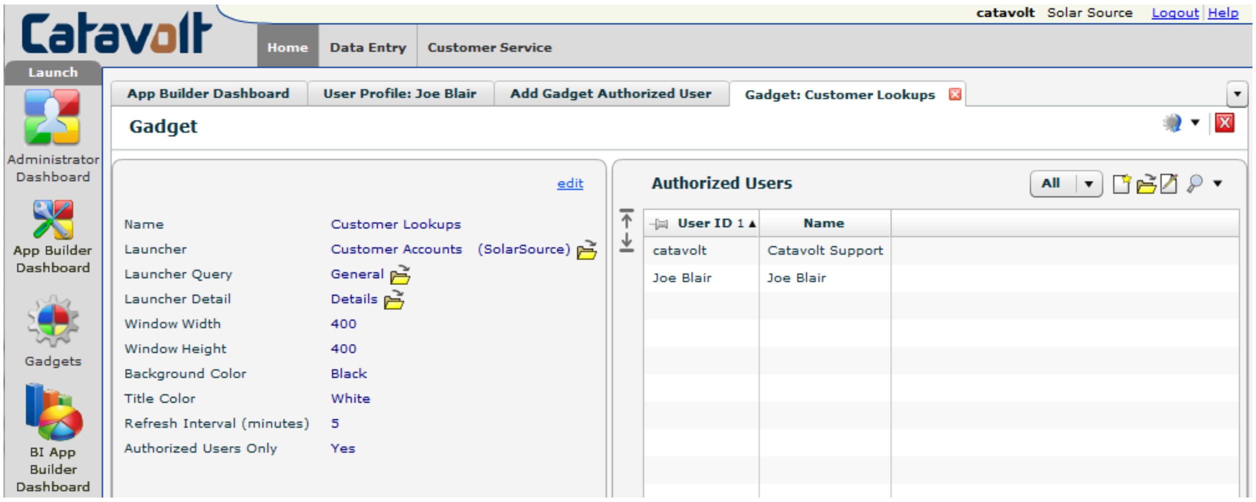
Selected

Gadget 1 ▲
Customer Lookups

Figure 14: The add Gadget Authorized User view.

You can select a single or multiple gadgets and press the Add button to add them to the User Profile.

After the Gadget Authorized User has been created, opening the Authorized User record will display the details for the Gadget. See Chapter 8: Gadgets for more information about Gadgets.



Gadget

Name	Customer Lookups
Launcher	Customer Accounts (SolarSource)
Launcher Query	General
Launcher Detail	Details
Window Width	400
Window Height	400
Background Color	Black
Title Color	White
Refresh Interval (minutes)	5
Authorized Users Only	Yes

Authorized Users

User ID 1 ▲	Name
catavolt	Catavolt Support
Joe Blair	Joe Blair

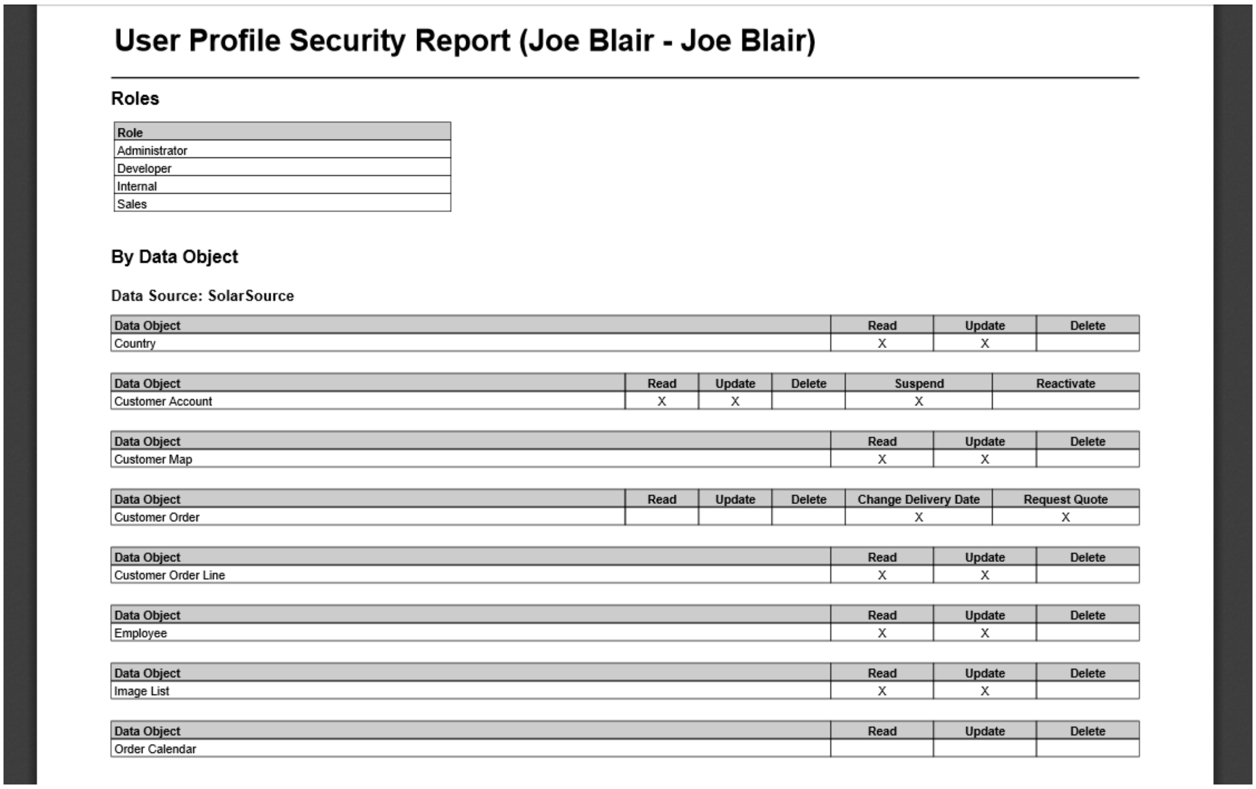
Figure 15: The Gadget details view.

Other User Profile Menu Options

Print Security Report

Selecting this menu option will generate a PDF document that describes in detail the authorizations granted to this User Profile. It will provide basic security information for the User Profile along with a list of Security Roles the User Profile is assigned to. In addition, a list of all Data Objects is included along with their "effective" authorizations. Effective authorizations are determined by combining all Security Roles that a user is a member of along with the permissions for each Security Role. Note that permissions are inclusive. If a user is a member of at least 1 Security Role with authorization to a Data Object, the user will be authorized.

The User Profile Security Report will look similar to the following picture:



User Profile Security Report (Joe Blair - Joe Blair)

Roles

Role
Administrator
Developer
Internal
Sales

By Data Object

Data Source: SolarSource

Data Object	Read	Update	Delete
Country	X	X	

Data Object	Read	Update	Delete	Suspend	Reactivate
Customer Account	X	X		X	

Data Object	Read	Update	Delete
Customer Map	X	X	

Data Object	Read	Update	Delete	Change Delivery Date	Request Quote
Customer Order				X	X

Data Object	Read	Update	Delete
Customer Order Line	X	X	

Data Object	Read	Update	Delete
Employee	X	X	

Data Object	Read	Update	Delete
Image List	X	X	

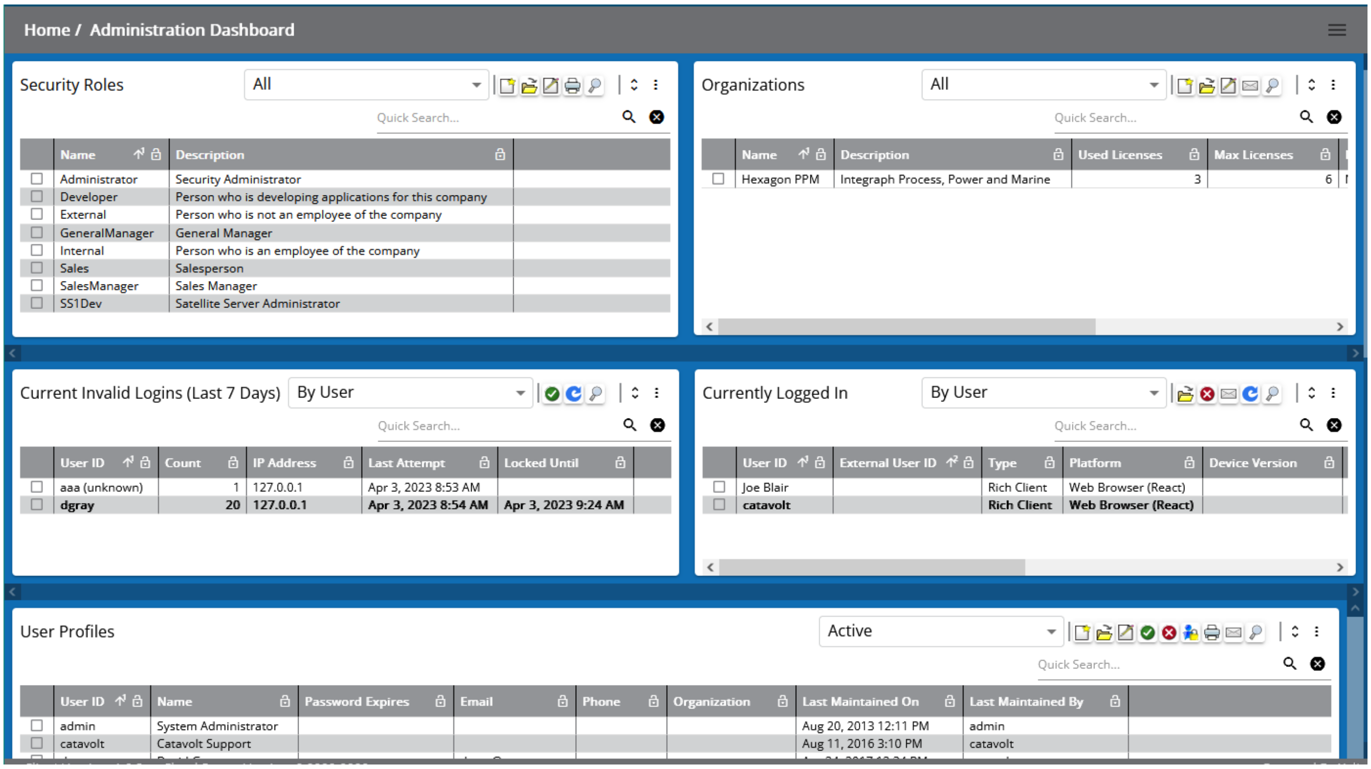
Data Object	Read	Update	Delete
Order Calendar			

Figure 16: A sample of a generated User Profile Security Report

See Chapter 10: Security for more information about object security in Xalt.

Administrator Dashboard

The Administrator Dashboard gives you a quick view into User Profiles for your system. Across the top you can see available Security Roles and a list of Organizations defined for your tenant. In the second row you can view a list of users that are currently having invalid login attempts as well as a list of users who are currently logged into Xalt. Along the bottom you can view User Profiles and Activity Log (user logins and maintenance activity). There are also two menu options in the top right to show User Activity and a list of Inactive Users (both by date).



Security Roles — All

	Name	Description
□	Administrator	Security Administrator
□	Developer	Person who is developing applications for this company
□	External	Person who is not an employee of the company
□	GeneralManager	General Manager
□	Internal	Person who is an employee of the company
□	Sales	Salesperson
□	SalesManager	Sales Manager
□	SS1Dev	Satellite Server Administrator

Organizations — All

	Name	Description	Used Licenses	Max Licenses
□	Hexagon PPM	Integraph Process, Power and Marine	3	6

Current Invalid Logins (Last 7 Days) — By User

	User ID	Count	IP Address	Last Attempt	Locked Until
□	aaa (unknown)	1	127.0.0.1	Apr 3, 2023 8:53 AM	
□	**dgray**	**20**	**127.0.0.1**	**Apr 3, 2023 8:54 AM**	**Apr 3, 2023 9:24 AM**

Currently Logged In — By User

	User ID	External User ID	Type	Platform	Device Version
□	Joe Blair		Rich Client	Web Browser (React)	
□	**catavolt**		**Rich Client**	**Web Browser (React)**	

User Profiles — Active

	User ID	Name	Password Expires	Email	Phone	Organization	Last Maintained On	Last Maintained By
□	admin	System Administrator					Aug 20, 2013 12:11 PM	admin
□	catavolt	Catavolt Support					Aug 11, 2016 3:10 PM	catavolt

Figure 17: The Administrator Dashboard view

Current Invalid Logins (Last 7 Days)

The Current Invalid Logins section shows you a live list of users who are currently unable to login. This list will show the following information:

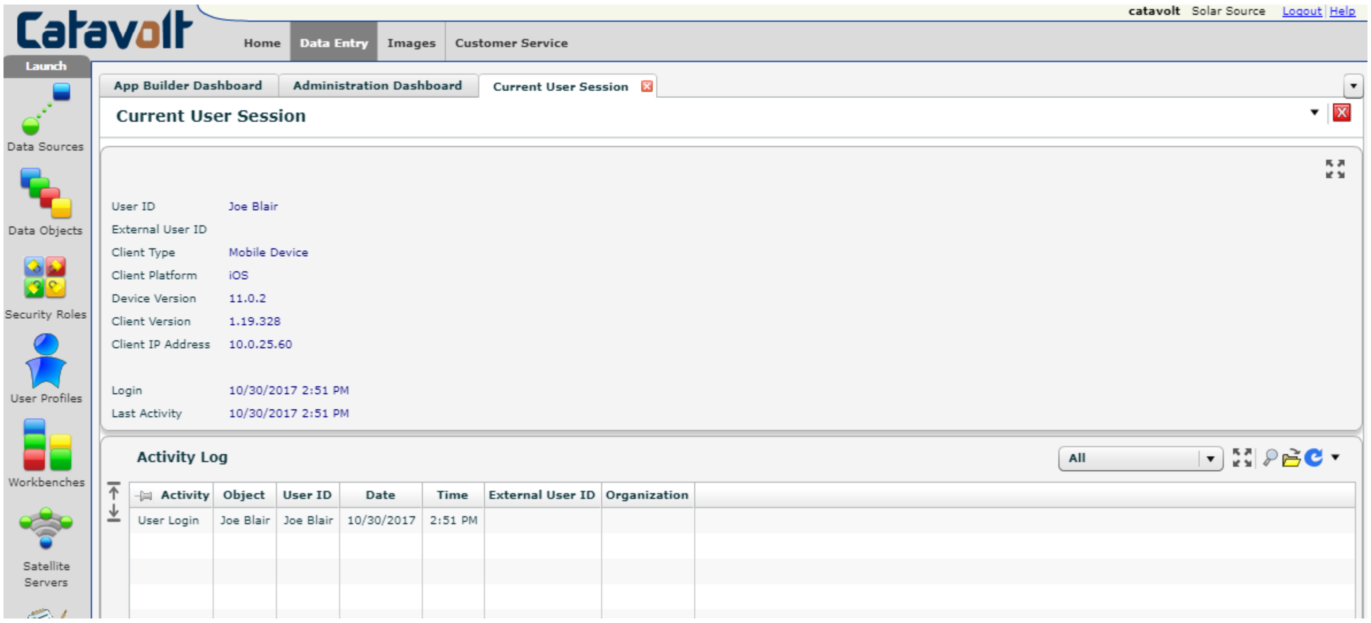
1. The User ID that is currently failing to login. If someone is attempting to login with an unknown User ID, **(unknown)** will be added to the end of the User ID. If this is an external User ID that you are validating using a custom login provider, **(external)** will be added to the end of the User ID.
2. The current consecutive login failure count
3. The client IP Address the last failed login attempt occurred from
4. The time of the last login attempt (this includes if the user attempts a login while locked out)
5. The time when the next login attempt for that user ID can occur (if the user is locked out)

Once the user successfully logs in, their record will be removed from this list. If a user makes 20 consecutive invalid login attempts, their account will be locked out for 30 minutes. If the user is currently locked out, the record will show in **bold**. There is an **Enable Login** action the Administrator can take to immediately cancel a lockout and let the selected user attempt a new login. In addition, if no login attempt has been made on the User ID for 7 days, the record will be automatically removed from this list and the user's invalid count will be set back to 0.

Note that we have also enhanced the Activity Log so that invalid login attempts will now be logged. We have added a Failed Logins view to allow you to easily view all failed logins together.

Currently Logged In

The Currently Logged In section shows you a list of active sessions for your tenant. Your own session in the Currently Logged In section will appear in bold. Other sessions are shown in normal font. You have the ability to Force End Sessions for other users currently logged in. You can also drill into a record to see more details of the user's session:



Current User Session

User ID Joe Blair
External User ID
Client Type Mobile Device
Client Platform iOS
Device Version 11.0.2
Client Version 1.19.328
Client IP Address 10.0.25.60

Login 10/30/2017 2:51 PM
Last Activity 10/30/2017 2:51 PM

Activity Log

Activity	Object	User ID	Date	Time	External User ID	Organization
User Login	Joe Blair	Joe Blair	10/30/2017	2:51 PM		

Figure 18: The Current User Session details view

You can see the following information about the session:

User ID specifies the Hexagon User ID for this user

External User ID specifies the External User ID (if using external authentication such as AD)

Client Type specifies the type of device. Values are Rich Client, Mobile Client, Gadget, and API.

Client Platform specifies the platform of the device. Values are Web Browser, iOS, and Android.

Device Version specifies the OS version of the device

Client Version specifies the version of the Hexagon application.

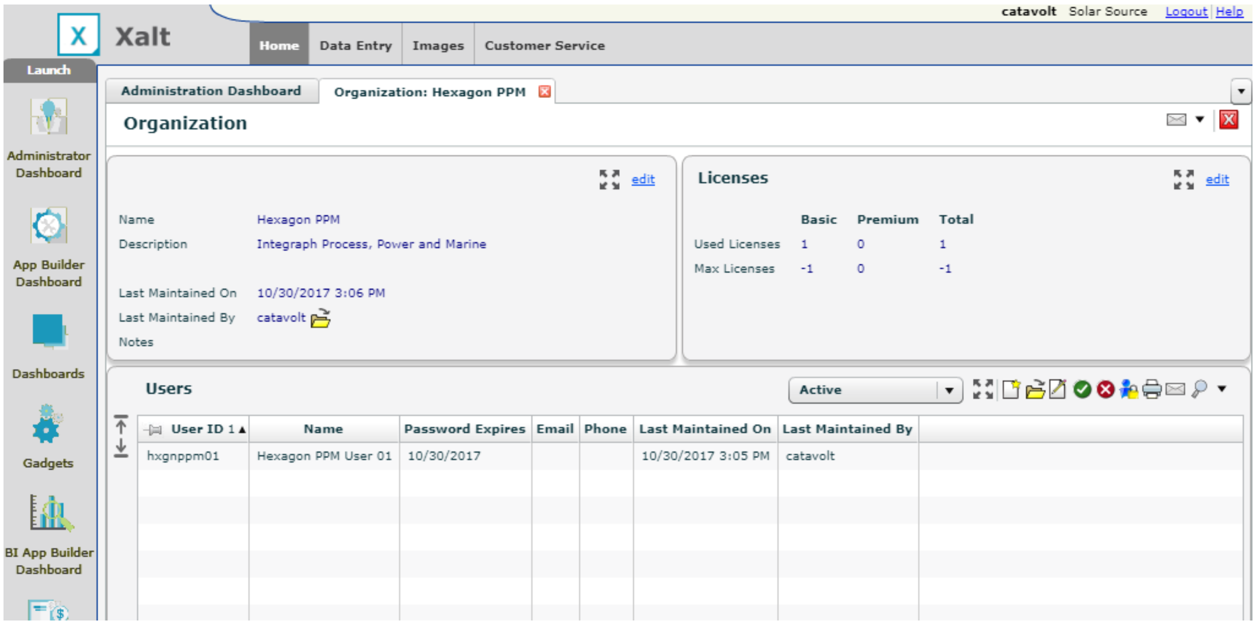
Client IP Address specifies the IP Address the device is connecting from

Login specifies the date/time the user logged in

Last Activity specifies the date/time the user last contacted the Xalt Cloud server.

In addition, you can view the Activity Log for that users' current session.

Organizations



Organization

Name	Hexagon PPM
Description	Integraph Process, Power and Marine
Last Maintained On	10/30/2017 3:06 PM
Last Maintained By	catavolt
Notes	

Licenses

	Basic	Premium	Total
Used Licenses	1	0	1
Max Licenses	-1	0	-1

Users

User ID 1 ▲	Name	Password Expires	Email	Phone	Last Maintained On	Last Maintained By
hxgnppm01	Hexagon PPM User 01	10/30/2017			10/30/2017 3:05 PM	catavolt

Figure 19: The Organization details view

Organizations allow you to create subgroups of Users within your tenant for the purpose of grouping for licensing, filtering activity, or allowing for local user administrators. Organization have the following values:

Name specifies the name of the Organization

Description specifies a description of the Organization

Max Licenses (Basic) – specifies the maximum number of User Profiles that can be assigned to this Organization as basic users. Setting this value to -1 will allow an unlimited number of User Profiles.

Max Licenses (Premium) – specifies the maximum number of User Profiles that can be assigned to this Organization as premium users.

Max Licenses (Total) – The sum of Basic and Premium Max Licenses (or -1 if Basic Licenses are set to -1). This value is display only.

Used Licenses (Basic) – specifies the number of User Profiles that are currently assigned to this Organization as basic users. Note that only Active User Profiles count against Used Licenses. Inactive User Profiles are not counted toward this total.

Used Licenses (Premium) – specifies the number of User Profiles that are currently assigned to this Organization as premium users. Note that only Active User Profiles count against Used Licenses. Inactive User Profiles are not counted toward this total.

Used Licenses (Total) – The sum of Basic and Premium Used Licenses. This value is display only.

Please note the following:

- Used Licenses (Premium) cannot exceed Max Licenses (Premium)
- Used Licenses (Basic) may exceed Max Licenses (Basic) but cannot exceed Max Licenses (Total). That is, a Basic License may use a Premium License if one is available.
- An Administrator may reduce Max Licenses to a level below Used Licenses. In this case, all existing users will continue to run, but no new users may be added until Used Licenses are reduced below Max Licenses.

Premium licensing is a feature added for some internal Hexagon tenants and is not applicable for our customers. The functionality is available for your own use.

Organization contains the standard menu actions along with a Send Email action that allows you to email all Users within an Organization at one time.

Organization Administrator

If a User Profile is assigned as an Organization Administrator, they will have the ability to do basic administration on other User Profiles within the Organization.

A new Launcher called "Organization Administration" has been added as an Available Launcher to be used on Workbenches for this purpose. However, for reasons to become clear below, it is expected that this Launcher will not be used on a Workbench under normal circumstances.

If an Organization Administrator logs into Xalt and they do not have this launcher on any of their Workbenches, an ad-hoc Workbench called "Admin" will be shown (Rich client only). The Admin Workbench will contain an Organization Administration launcher. Note that this is similar to the Recovery workbench for Admin/Developers.

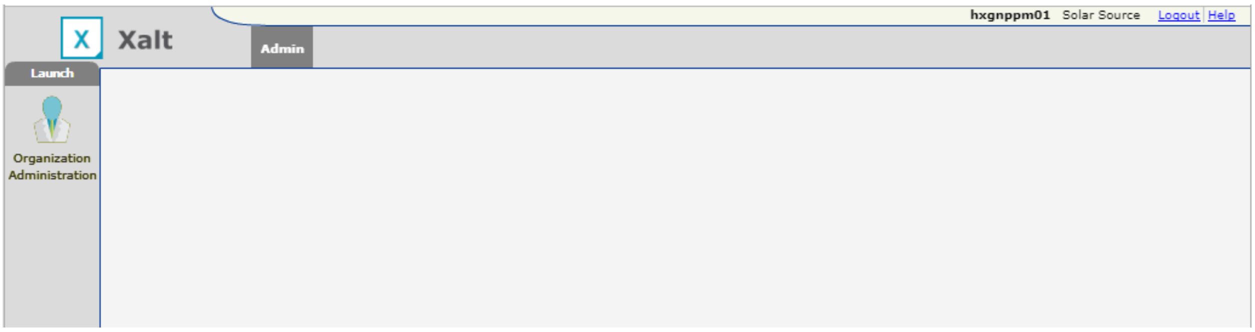
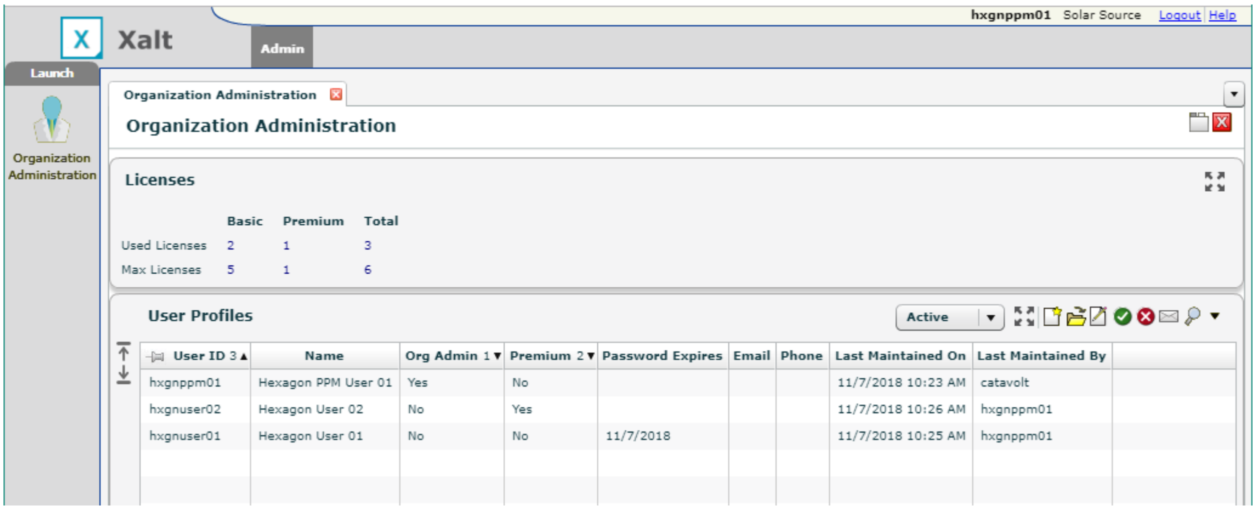

Xalt
hxgnppm01 Solar Source Logout | Help
Admin
Launch
Organization Administration

Figure 20: The Organization Admin ad-hoc Workbench

Clicking the launcher will show the list of User Profiles in the same Organization as the current user. Only Organization Administrators can see this list. You have the choice of Active, Inactive, and All user profiles within the Organization.


Xalt
hxgnppm01 Solar Source Logout | Help
Admin
Launch
Organization Administration
Organization Administration

Organization Administration

Licenses

	Basic	Premium	Total
Used Licenses	2	1	3
Max Licenses	5	1	6

User Profiles

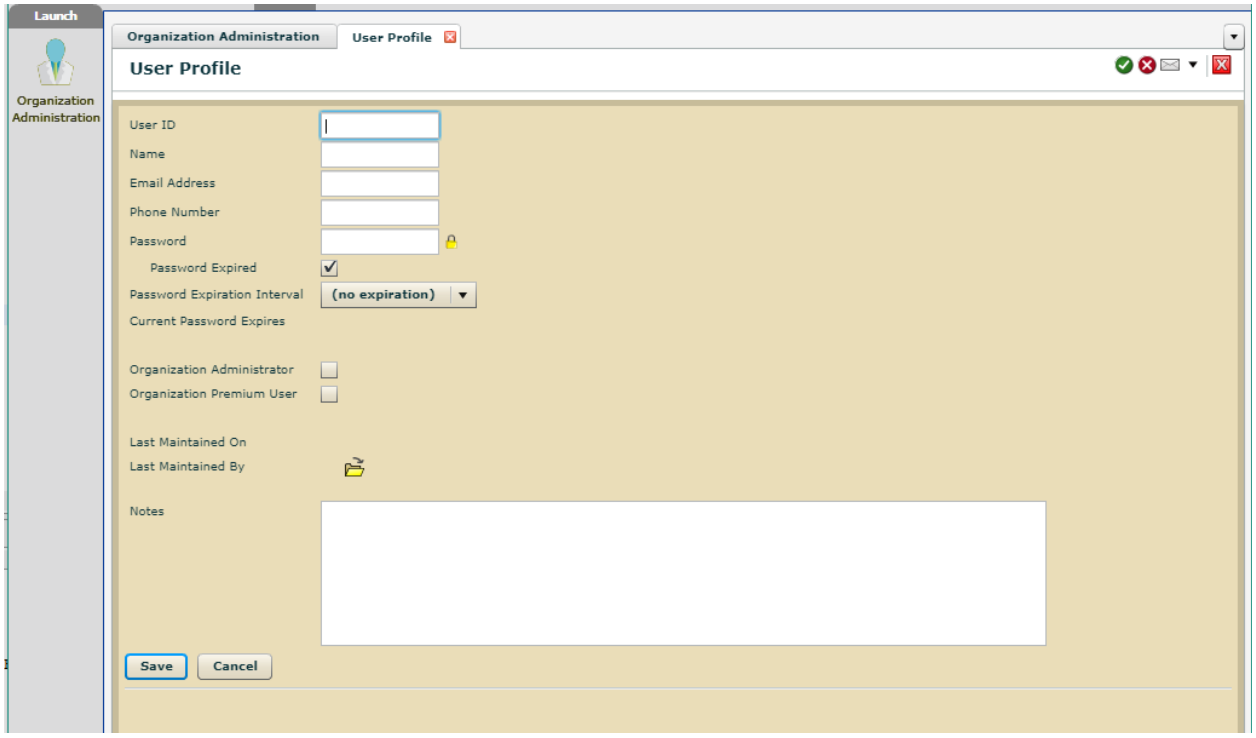
Active

User ID 3 ▴	Name	Org Admin 1 ▾	Premium 2 ▾	Password Expires	Email	Phone	Last Maintained On	Last Maintained By
hxgnppm01	Hexagon PPM User 01	Yes	No				11/7/2018 10:23 AM	catavolt
hxgnuser02	Hexagon User 02	No	Yes				11/7/2018 10:26 AM	hxgnppm01
hxgnuser01	Hexagon User 01	No	No	11/7/2018			11/7/2018 10:25 AM	hxgnppm01

Figure 21: The Organization Administration details view

The following menu options are available. These are a subset of the normal User Profile menu actions:

- New
- Open
- Edit
- Activate
- Suspend
- Send Email
- Refresh
- Search
- Export



User ID
Name
Email Address
Phone Number
Password
Password Expired ☑
Password Expiration Interval (no expiration)
Current Password Expires
Organization Administrator □
Organization Premium User □
Last Maintained On
Last Maintained By
Notes

Figure 22: The Organization Administrator User Profile details view

The New option allows the user to create a new User Profile for the User's Organization. The admin will be able to fill in a small subset of values. Notice that most values like Workbenches, Security Roles, etc. are not available. All options other than those above will be copied from the Organization Admin user performing the create. Under the covers, creating a new user causes the Organization Admin user to copy his own User Profile to the new user and then overlay the above values over the top. This is to enforce the design that an Organization Administrator user is just a regular user with the ability to create and administer other similar users.

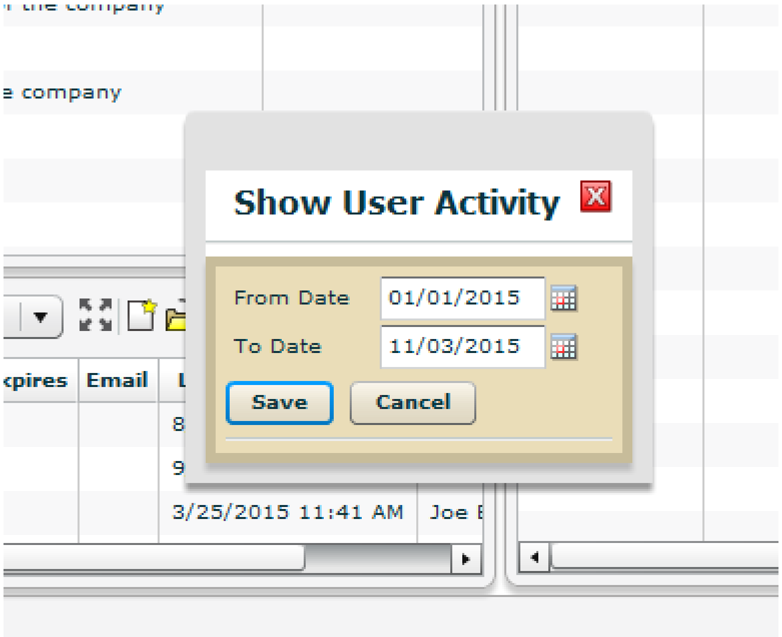
The following differences may occur during Create:

- If the Organization Administrator has Administrator or Developer security, these will be stripped out of the new user
- The new user will be forced to have (Hexagon) Authentication regardless of what the Organization Administrator user has

If the tenant has automatic email notifications for User Profile create and password reset, these will work as normal for these User Profiles.

Show User Activity

When choosing the Show User Activity menu option, you are presented with a Date range to show activity for. By default, the current YTD range will be displayed.



Show User Activity

From Date 01/01/2015

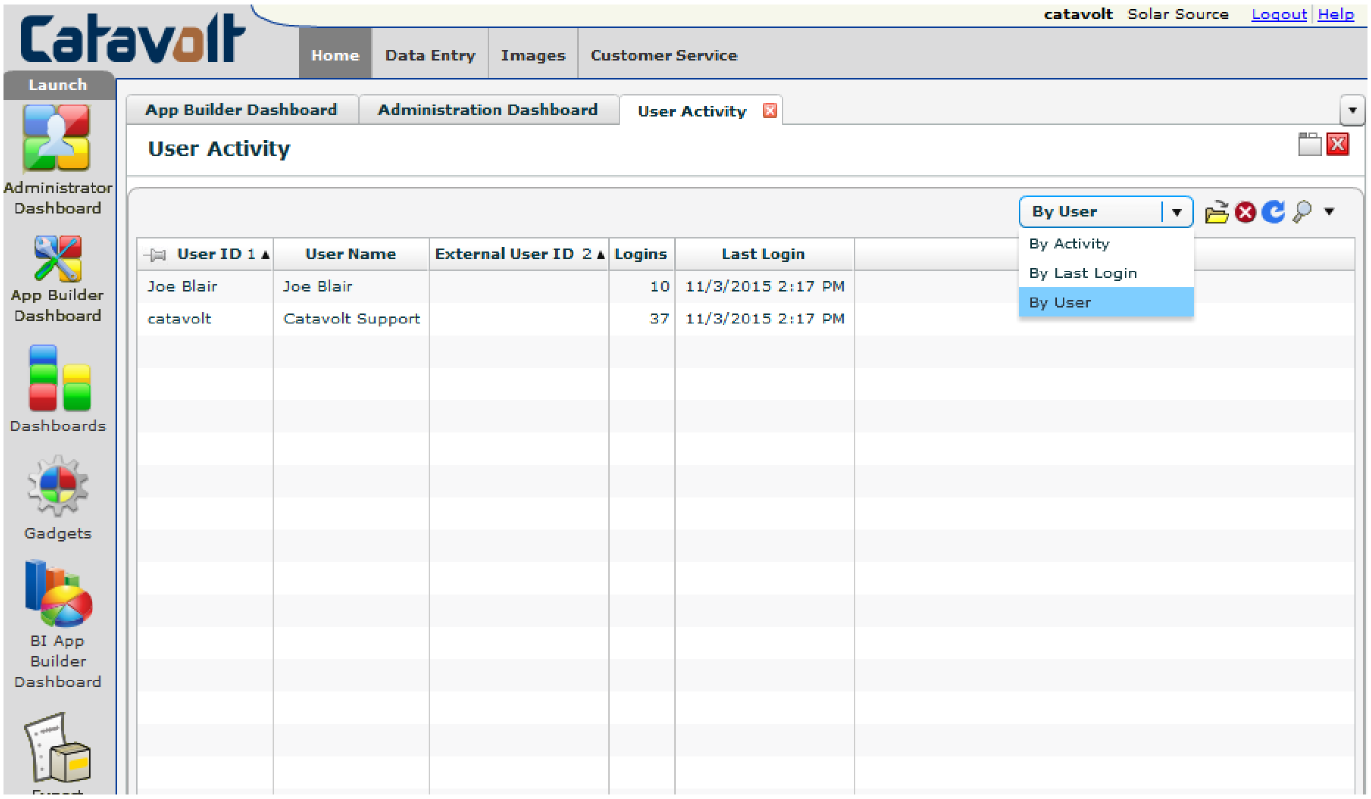
To Date 11/03/2015

Figure 23: The Show User Activity range view

You will be presented with a window showing all users, internal and external, who have logged into Xalt during that time period, along with the number of logins made during the time period and the time of last login. You have 3 built-in options on how the list is sorted:

- By Activity (# of Logins descending)
- By Last Login (Last Login descending)
- By User (User ID ascending)

In addition, there is a Suspend User menu option. You can select one or more User Profiles and suspend them so that they cannot login (Note only internal users can be suspended using this menu option). User Profiles that have been suspended can be reactivated from the User Profiles window.

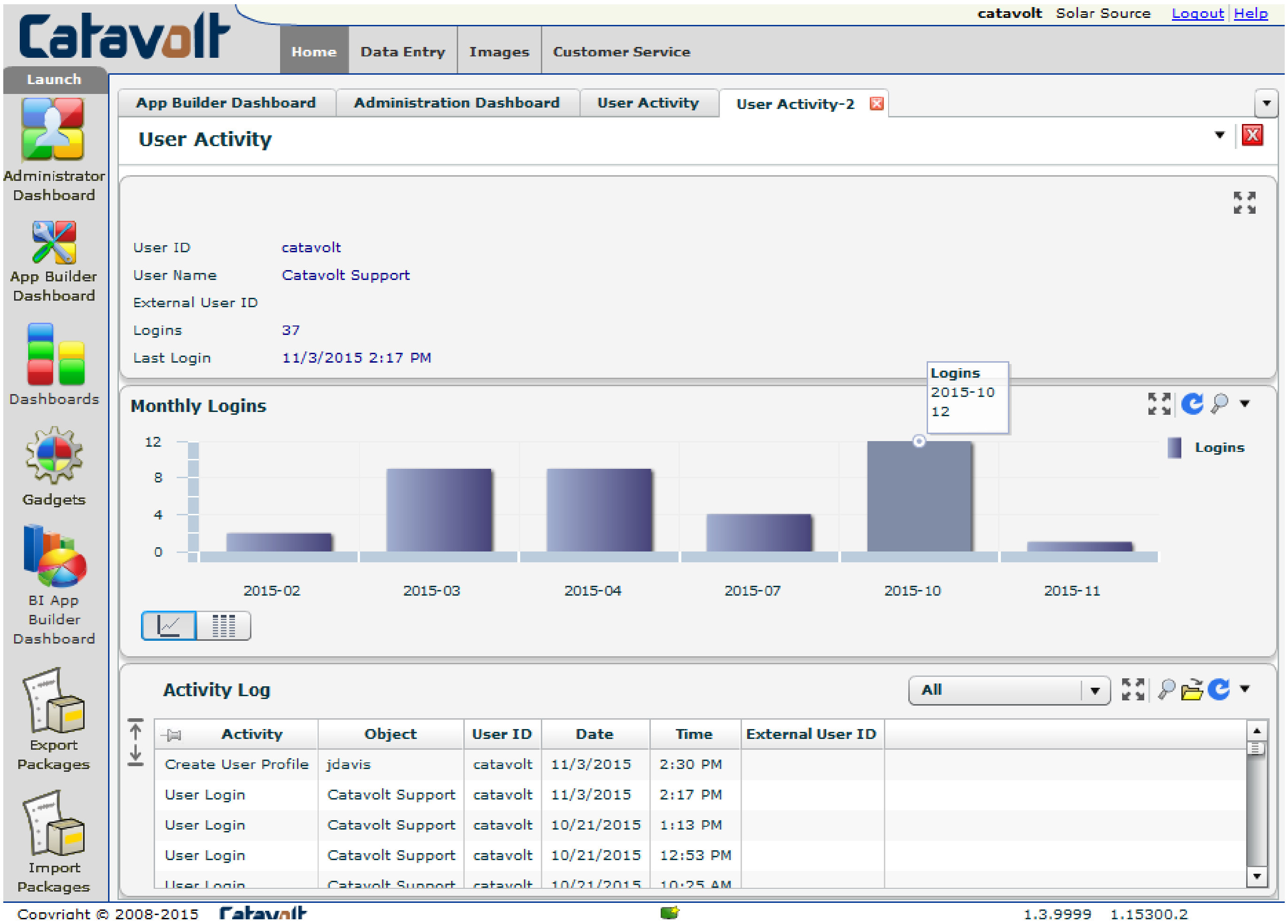


User Activity

User ID 1 ▲	User Name	External User ID 2 ▲	Logins	Last Login
Joe Blair	Joe Blair		10	11/3/2015 2:17 PM
catavolt	Catavolt Support		37	11/3/2015 2:17 PM

Figure 24: The User Activity list view

If you double click a record to see more detail, you will be presented with a window breaking out the number of logins per month over the selected period along with the Activity Log of the user over the selected period.



User Activity

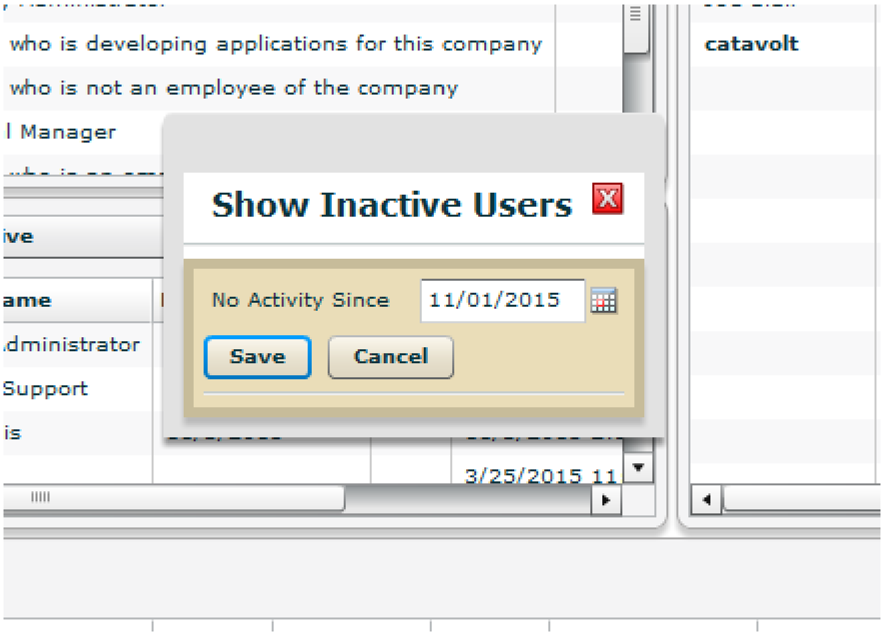
User ID	catavolt
User Name	Catavolt Support
External User ID	
Logins	37
Last Login	11/3/2015 2:17 PM

Activity	Object	User ID	Date	Time	External User ID
Create User Profile	jdavis	catavolt	11/3/2015	2:30 PM	
User Login	Catavolt Support	catavolt	11/3/2015	2:17 PM	
User Login	Catavolt Support	catavolt	10/21/2015	1:13 PM	
User Login	Catavolt Support	catavolt	10/21/2015	12:53 PM	
User Login	Catavolt Support	catavolt	10/21/2015	10:25 AM	

Figure 25: The User Activity details view

Show Inactive Users

When choosing the Show Inactive Users menu option, you are presented with a dialog to show inactivity since the selected cutoff date. By default, the 1st day of the current month will be displayed.



Show Inactive Users

No Activity Since 11/01/2015

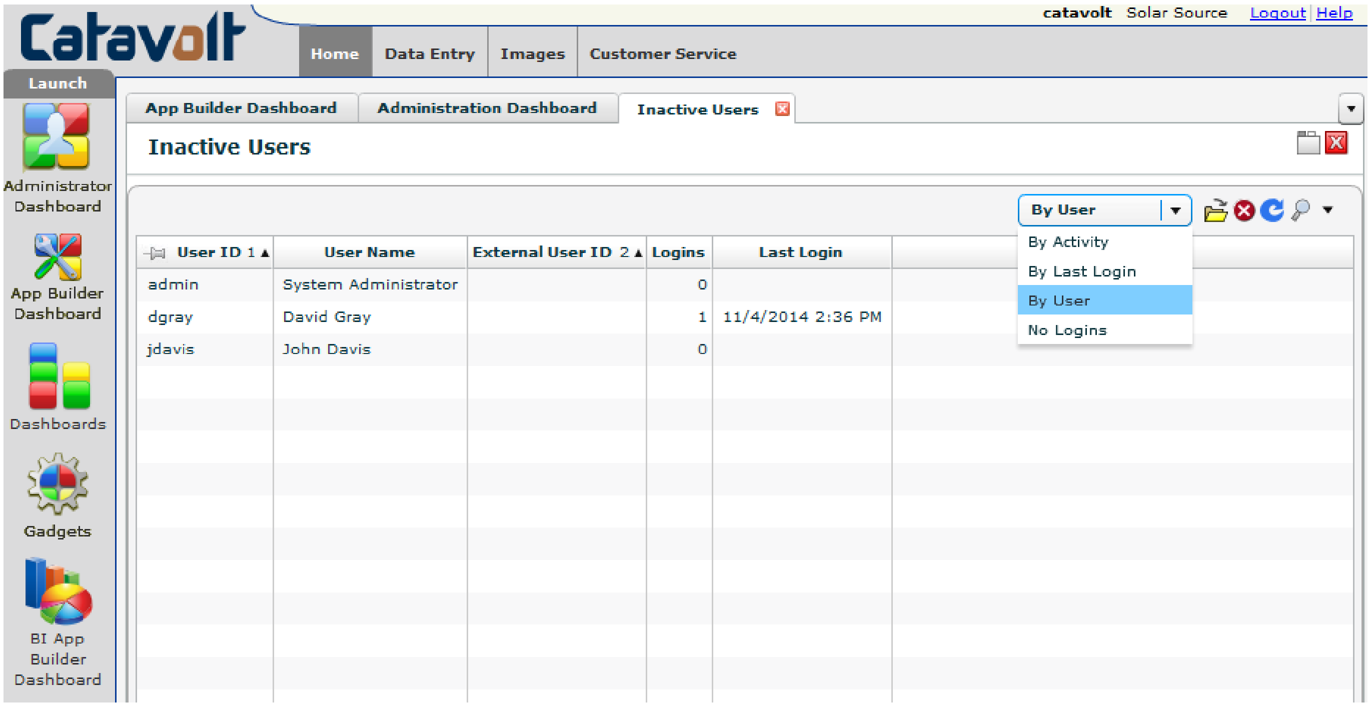
who is developing applications for this company
who is not an employee of the company
l Manager
catavolt
ive
ame
dministrator
Support
is
3/25/2015 11
Save
Cancel

Figure 26: The Show Inactive Users date view

You will be presented with a window showing all users, internal and external, who have not logged into Xalt since the chosen date, along with the total number of logins and the time of last login. You have 4 built-in options on how the list is sorted:

- By Activity (# of Logins descending)
- By Last Login (Last Login descending)
- By User (User ID ascending)
- No Logins (only show users who have never logged into Xalt)

In addition, there is a Suspend User menu option. You can select one or more User Profiles and suspend them so that they cannot login (Note only internal users can be suspended using this menu option). User Profiles that have been suspended can be reactivated from the User Profiles window.

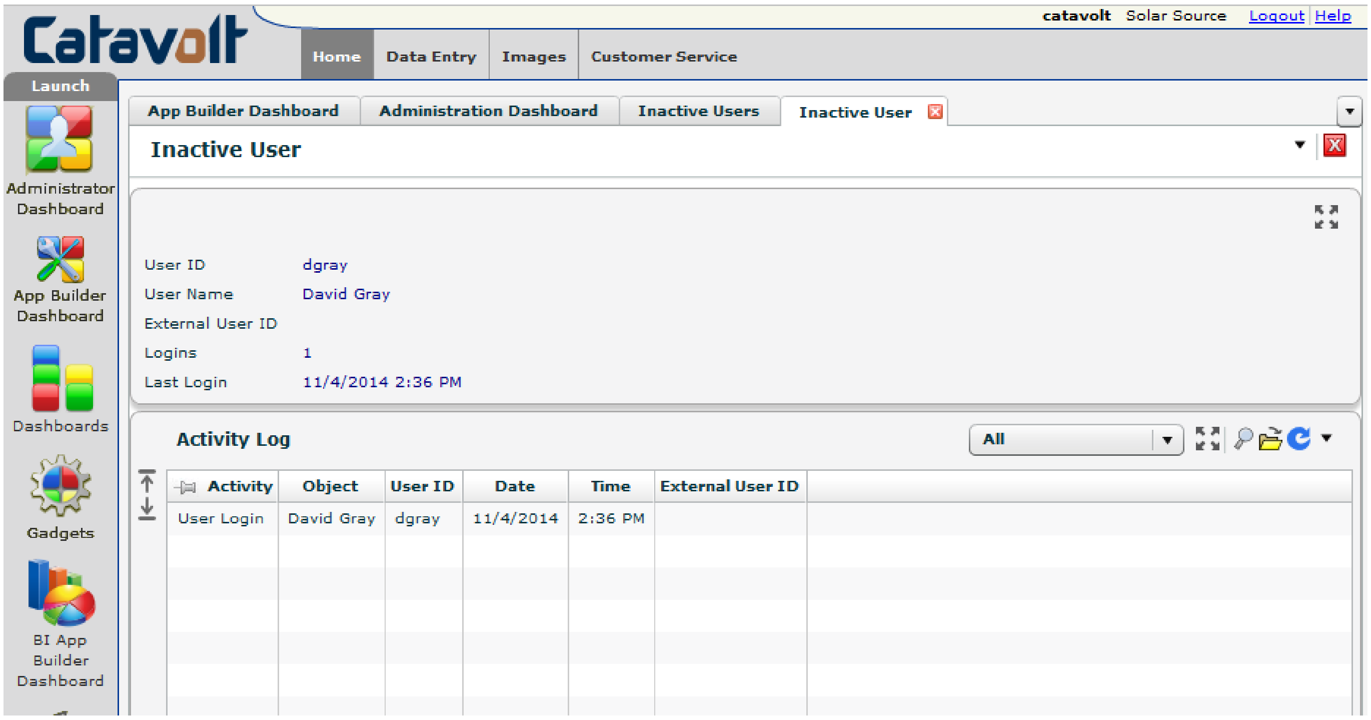


Inactive Users

User ID 1 ▲	User Name	External User ID 2 ▲	Logins	Last Login
admin	System Administrator		0	
dgray	David Gray		1	11/4/2014 2:36 PM
jdavis	John Davis		0	

Figure 27: The Inactive Users list view

If you double click a record to see more detail, you will be presented with the Activity Log of the user.



User ID dgray
User Name David Gray
External User ID
Logins 1
Last Login 11/4/2014 2:36 PM

Activity	Object	User ID	Date	Time	External User ID
User Login	David Gray	dgray	11/4/2014	2:36 PM	

Figure 28: The Inactive User details view

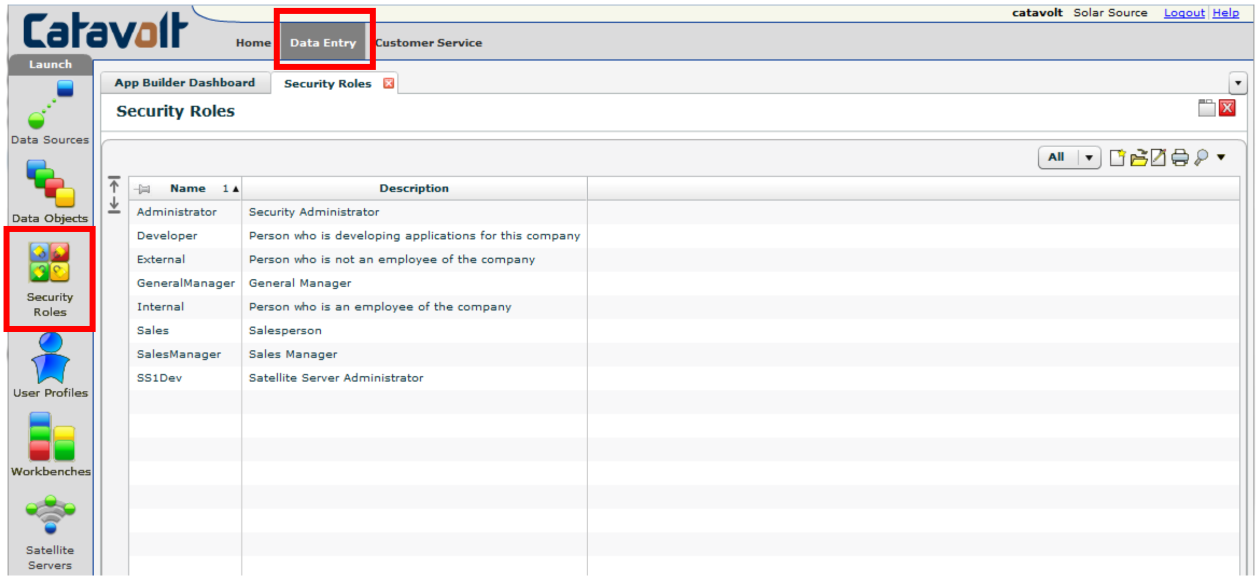
Chapter 10: Security

Chapter Summary

Xalt | Mobility employs a Role-based security scheme. All operations in Xalt are authorized to be performed by one or more Security Roles. Likewise, User Profiles can be members of one or more Security Roles. When a user attempts an operation, Xalt security checks the user's effective permissions to see if the user is authorized to perform the operation. Hexagon uses an inclusive method of determining a user's effective permissions. If the user is a member of at least one Security Group that has authorization to the operation, then the user is authorized.

Security Roles

Security Roles are the building blocks of Xalt Security. You can access the list of Security Roles by clicking on the Security Roles launcher in the Data Entry Workbench.



Security Roles

Name 1 ▲	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	Satellite Server Administrator

Figure 1: The Security Roles list view with the Workbench and Launcher highlighted

Standard Roles

There are 4 standard roles that are shipped with Xalt | Mobility (Administrator, Developer, Internal, and External). These roles have special meanings within the system and cannot be removed.

The **Administrator** role has authorization to perform most setup and security operations as well as access to runtime data. Administrators can access and manipulate the following objects:

- Activity Log
- BI Dashboards (read-only)
- BI Data Sets (read-only)
- Data Objects (read-only)

- Data Object Permissions
- Data Object Action Permissions
- Data Sources
- Default Properties
- Gadgets (read-only)

- Gadget Authorized Users
- Launchers
- Connector gateways
- Security Roles
- User Profiles
- Workbenches
- Workbench Users

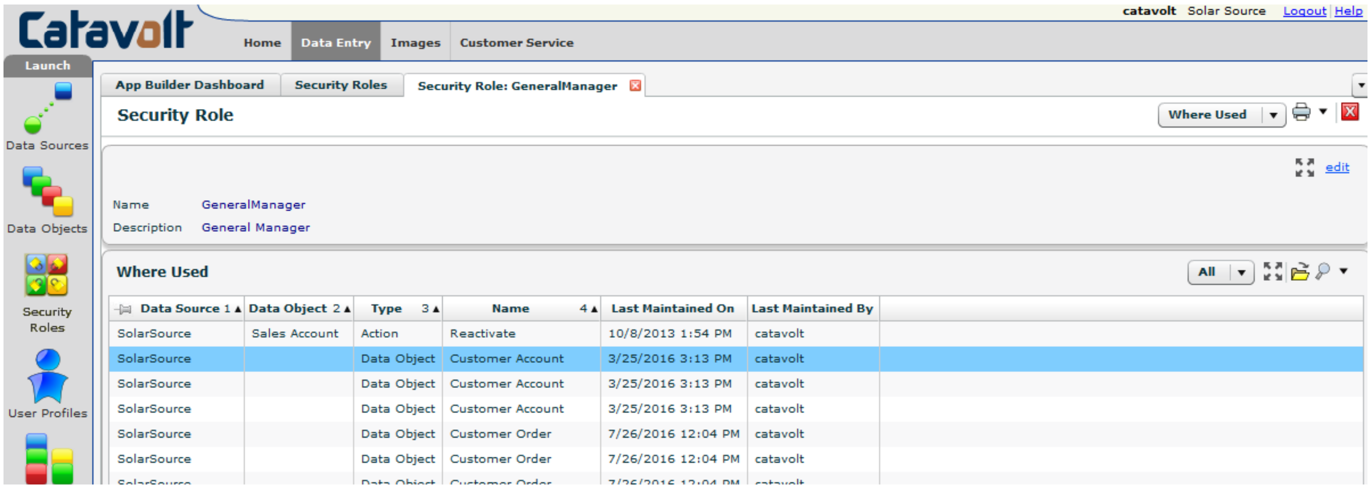
The **Developer** role has authorization to perform most setup and application-building operations as well as access to runtime data. Developers can access and manipulate the following objects:

- BI Dashboards
- BI Data Sets
- Data Objects
- Data Object Permissions (read-only)
- Data Object Action Permissions (read-only)
- Data Sources
- Default Properties
- Gadgets
- Launchers
- Connector gateways
- Connector gateway Permissions (read-only)
- Security Roles (read-only)
- User Profiles (read-only)
- Workbenches
- Workbench Users (read-only)

The **Internal** role indicates a person inside the organization (employee, manager, etc.) and has access to runtime data.

The **External** role indicates a person outside the organization (customer, supplier, etc.) and has access to runtime data.

When you display the definition for a Security Role, there are 2 Details that you can select from. The Default Detail shows you all the information about the Security Role. The Used By Detail shows a Where Used list of objects that are currently using this Security Role.



Security Role

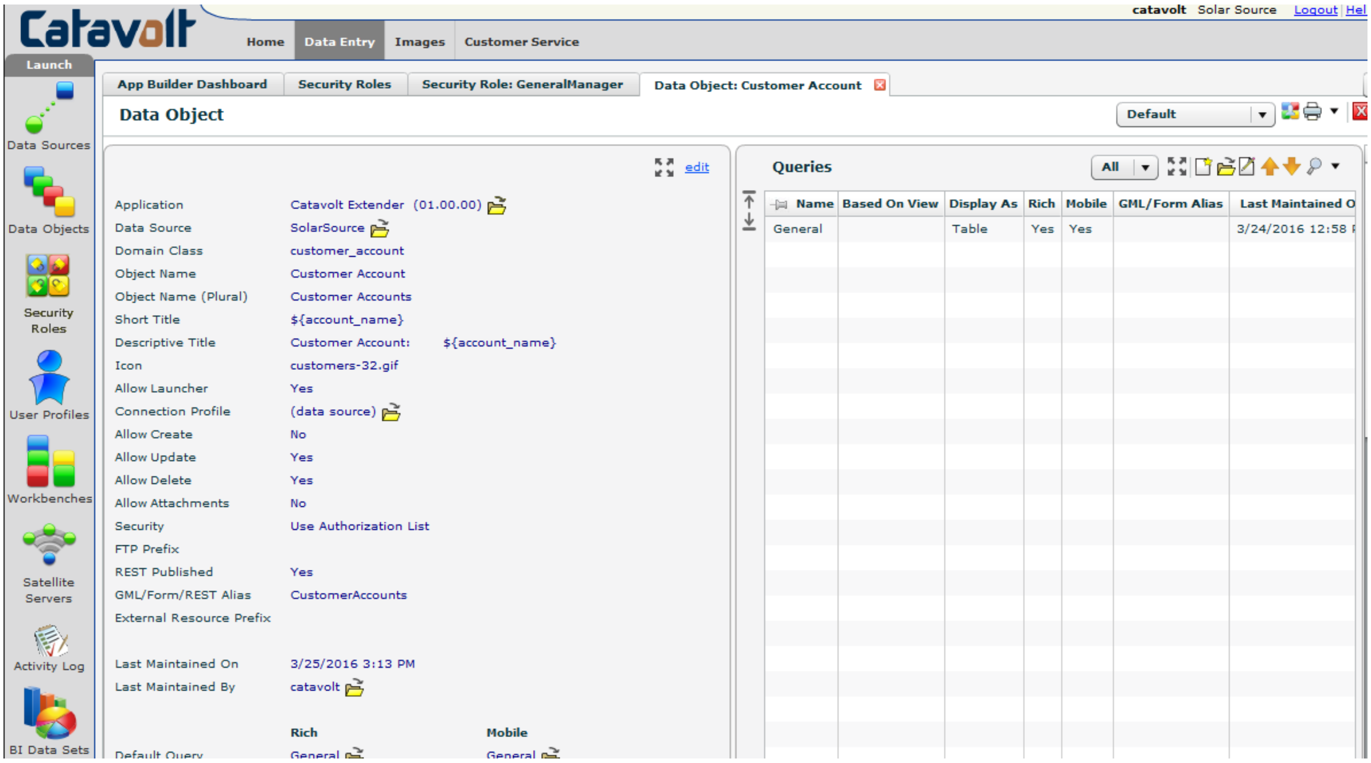
Name GeneralManager
Description General Manager

Where Used

Data Source 1 ▲	Data Object 2 ▲	Type 3 ▲	Name 4 ▲	Last Maintained On	Last Maintained By
SolarSource	Sales Account	Action	Reactivate	10/8/2013 1:54 PM	catavolt
SolarSource		Data Object	Customer Account	3/25/2016 3:13 PM	catavolt
SolarSource		Data Object	Customer Account	3/25/2016 3:13 PM	catavolt
SolarSource		Data Object	Customer Account	3/25/2016 3:13 PM	catavolt
SolarSource		Data Object	Customer Order	7/26/2016 12:04 PM	catavolt
SolarSource		Data Object	Customer Order	7/26/2016 12:04 PM	catavolt
SolarSource		Data Object	Customer Order	7/26/2016 12:04 PM	catavolt

Figure 2: Security Role 'Used By' details

Opening a Where Used record will take you directly to the object using the Security Role:



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_account
Object Name	Customer Account
Object Name (Plural)	Customer Accounts
Short Title	${account_name}
Descriptive Title	Customer Account: ${account_name}
Icon	customers-32.gif
Allow Launcher	Yes
Connection Profile	(data source)
Allow Create	No
Allow Update	Yes
Allow Delete	Yes
Allow Attachments	No
Security	Use Authorization List
FTP Prefix	
REST Published	Yes
GML/Form/REST Alias	CustomerAccounts
External Resource Prefix	
Last Maintained On	3/25/2016 3:13 PM
Last Maintained By	catavolt

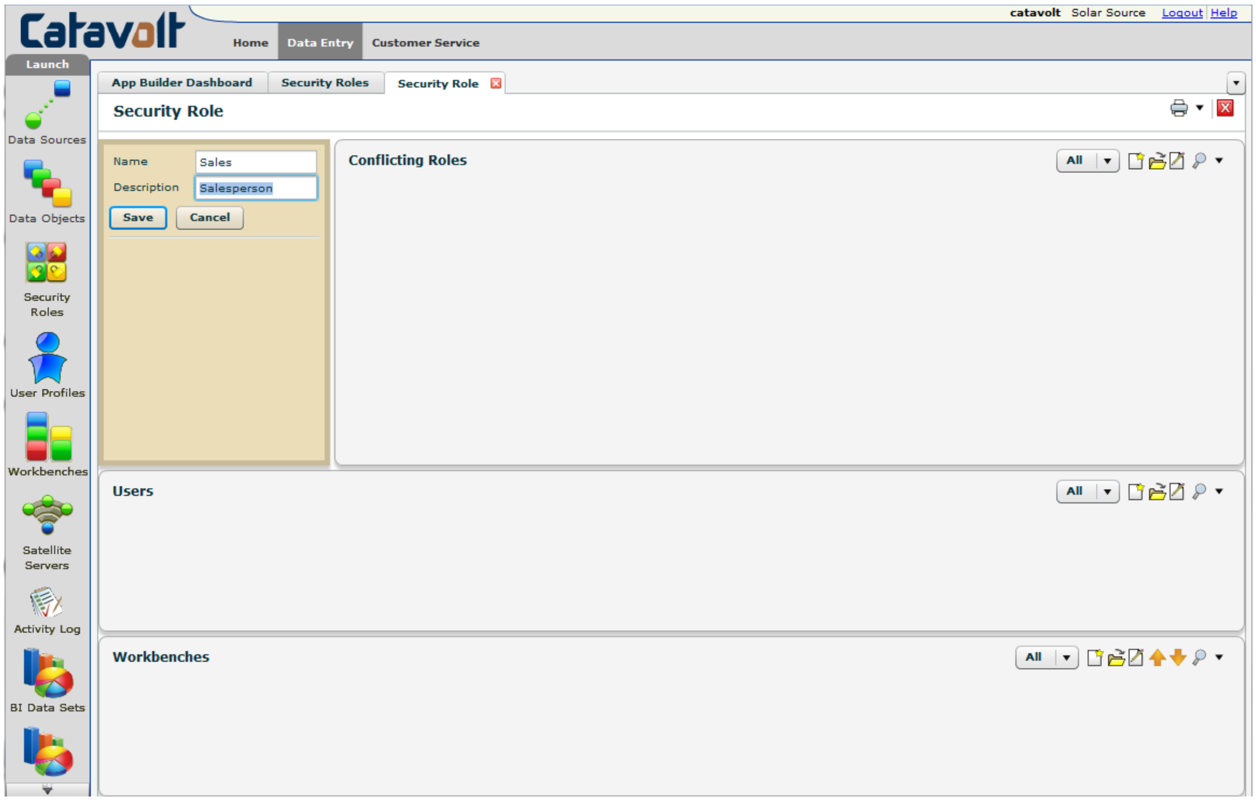
	Rich	Mobile
Default Query	General	General

Queries

Name	Based On View	Display As	Rich	Mobile	GML/Form Alias	Last Maintained O
General		Table	Yes	Yes		3/24/2016 12:58

Figure 3: Security Role Where Used

Creating Security Roles



Name: Sales

Description: Salesperson

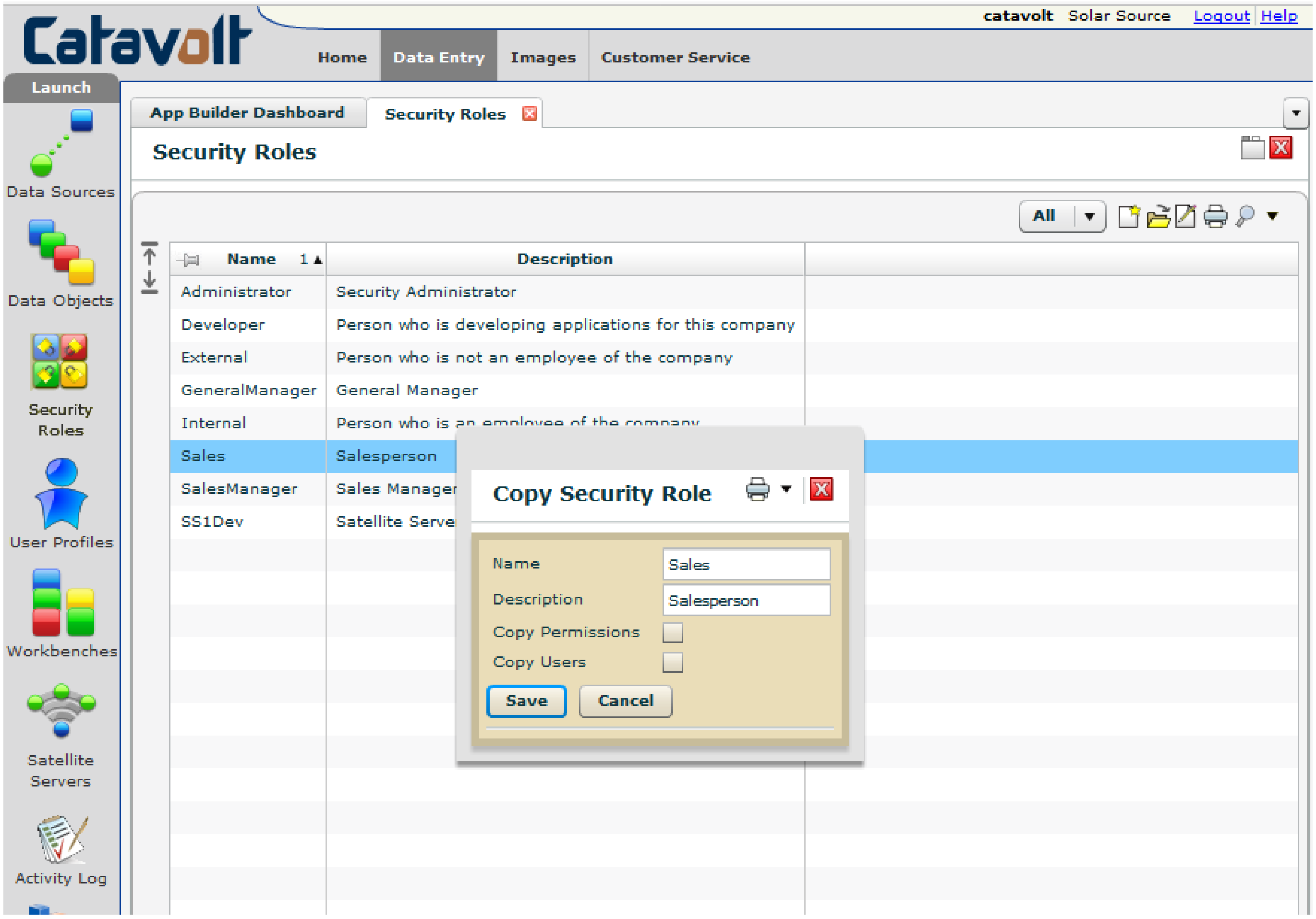
Figure 4: The create Security Role details

When creating a Security Role, you must specify a Name. This is the value that identifies the Security Role.

Description provides a brief description of the Security Role.

Copying Security Roles

You may have instances where you need to make a copy of an existing Security Role to deploy to a different set of users or to have a different set of permissions. You can select the Copy menu option to accomplish this. When Copying a Security Role, you will be prompted to supply a new Name and Description. An exact copy of this Security Role along with its components will be made.



Security Roles

Name 1 ▲	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson
SalesManager	Sales Manager
SS1Dev	Satellite Serve

Copy Security Role

Name	Sales
Description	Salesperson
Copy Permissions	□
Copy Users	□

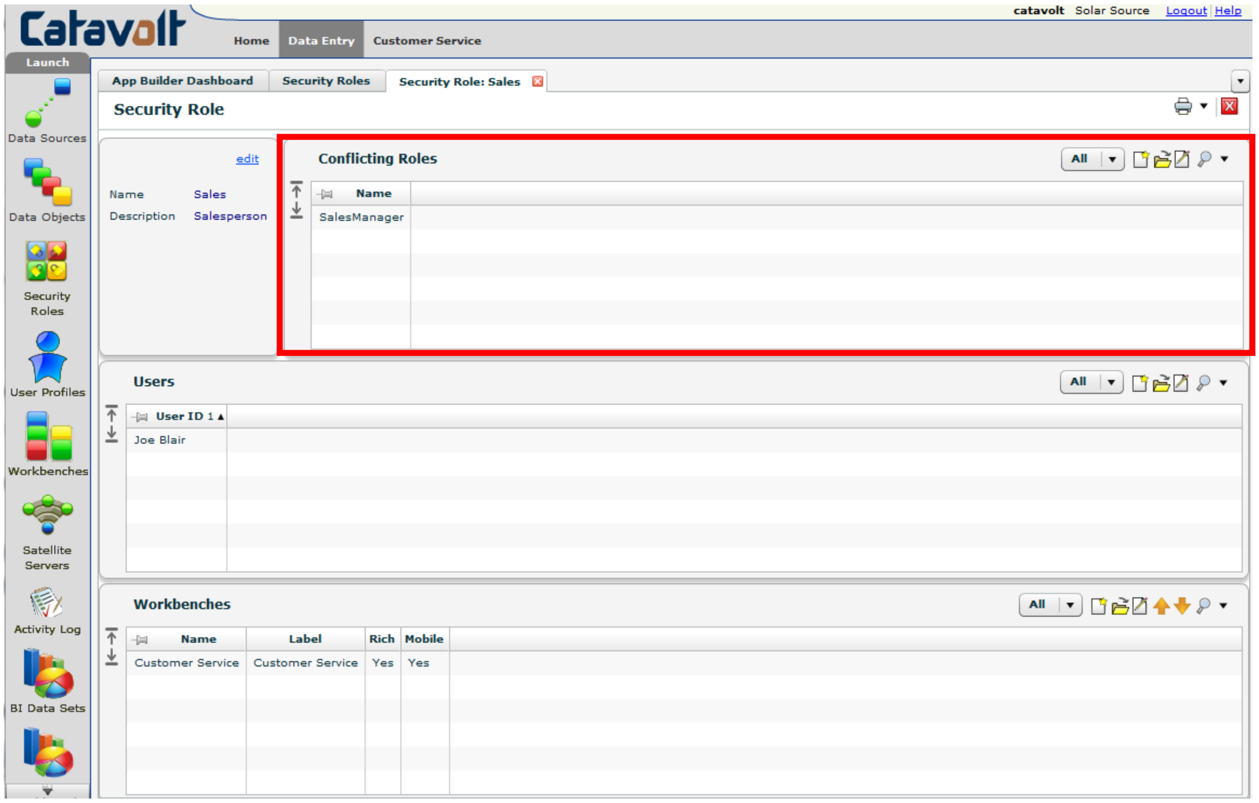
Figure 5: The copy Security Role dialog

Copy Permissions specifies whether to copy the permissions (Data Object, Actions, and Connector gateway) assigned to the existing Security Role.

Copy Users specifies whether to copy the users assigned to the existing Security Role

Conflicting Roles

When defining a Security Role, you can also identify one or more other Security Roles as Conflicting Roles. Conflicting Roles are roles that are incompatible with each other. A User Profile cannot be a member of 2 Security Roles that are conflicting (for example, a person cannot be a Salesperson and a Sales Manager at the same time). Defining Conflicting Roles is optional and is typically used by companies to enforce internal restrictions and oversight (e.g. a person cannot submit requests and also approve requests).



Name Sales
Description Salesperson

Conflicting Roles

Name
SalesManager

Users

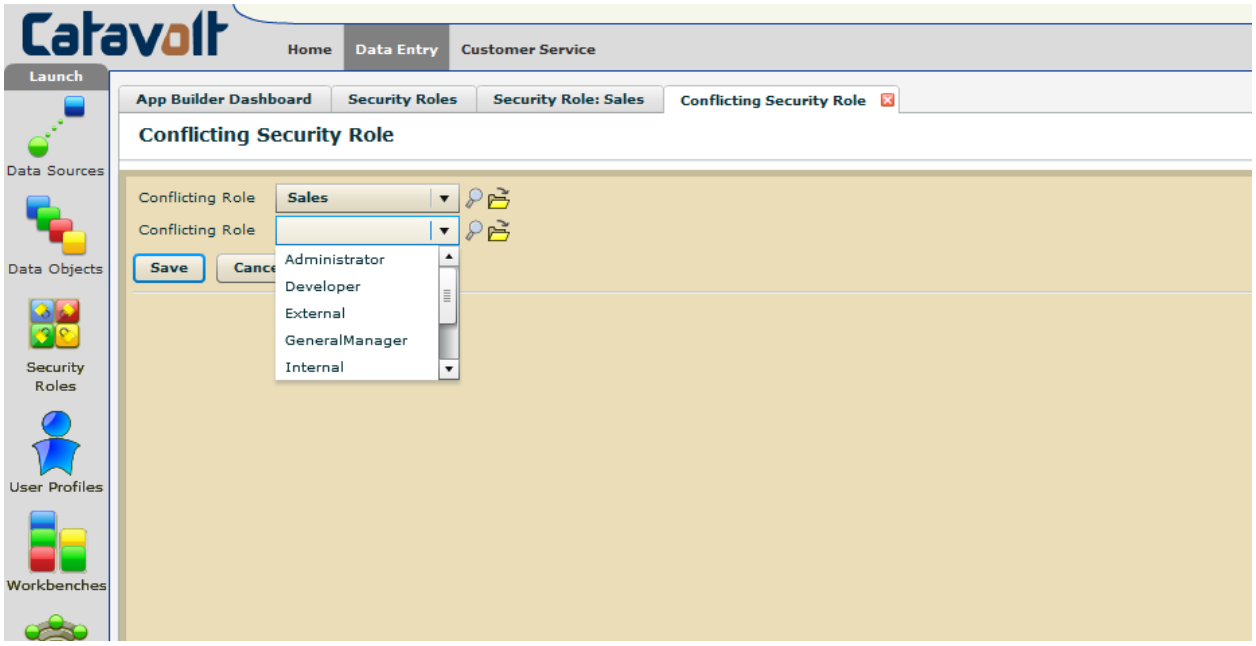
User ID 1 ▲
Joe Blair

Workbenches

Name	Label	Rich	Mobile
Customer Service	Customer Service	Yes	Yes

Figure 6: The Security Role details with the Conflicting Roles section highlighted

When adding Conflicting Roles, you will be presented with the following dialog



Conflicting Security Role

Conflicting Role: Sales

Conflicting Role:

Figure 7: The add Conflicting Role view.

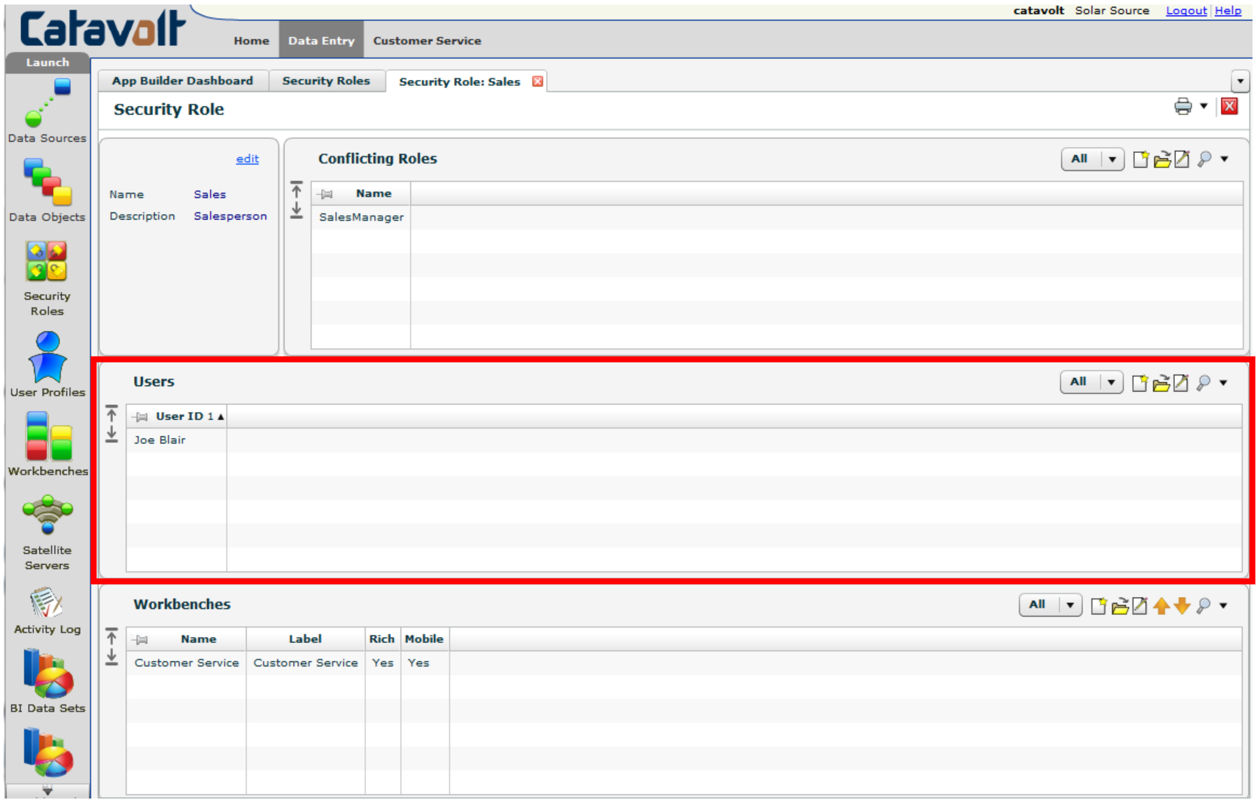
The first Conflicting Role will be filled in with the Security Role that you are viewing. Select the second Conflicting Role that conflicts with the first. Note that if any User Profiles are currently members of both Security Roles, you will

receive an error message and will not be able to make these roles conflicting until you remove the User Profile from one of the Security Roles.

After the Conflicting Role has been created, opening the selected record will display the details for the Security Role.

Security Role Assigned Users

Once a Security Role has been created, you can assign users to be members of the Role in the Users section.



Name Sales

Description Salesperson

Conflicting Roles

Name
SalesManager

Users

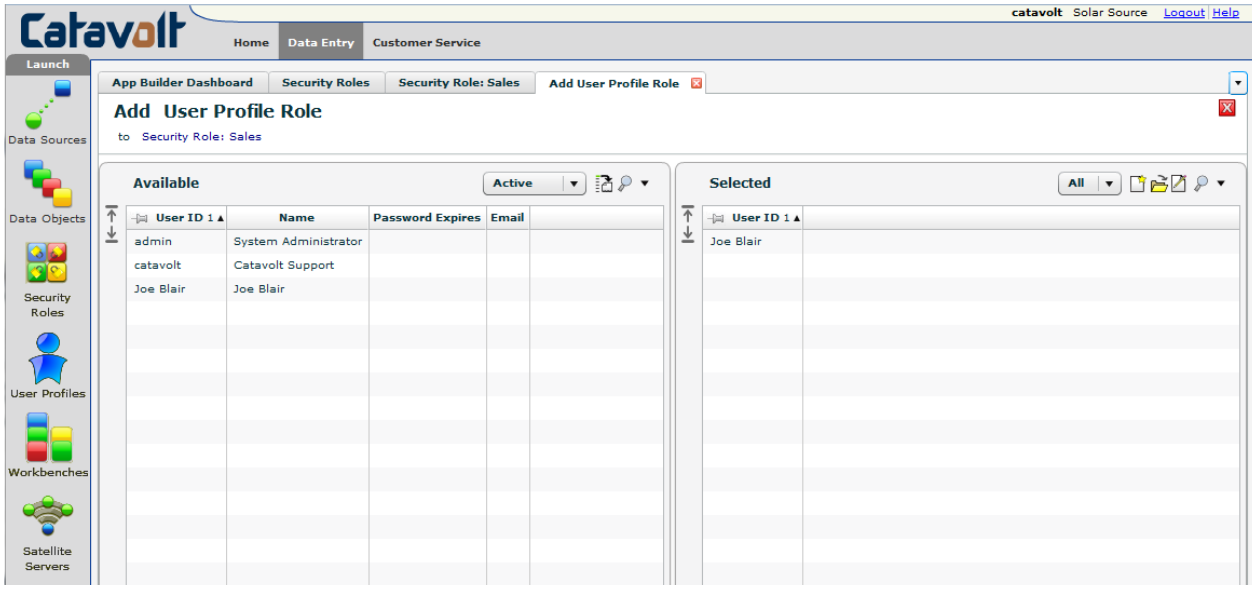
User ID 1 ▲
Joe Blair

Workbenches

Name	Label	Rich	Mobile
Customer Service	Customer Service	Yes	Yes

Figure 8: The Security Role details with the Users section highlighted

When adding User Security Roles, you will be presented with two lists. The Available Users list shows all User Profiles. The Selected Users list shows the users that will be members of the Security Role.



Add User Profile Role

to Security Role: Sales

Available

User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Selected

User ID 1 ▲
Joe Blair

Figure 9: The add User Security Role view.

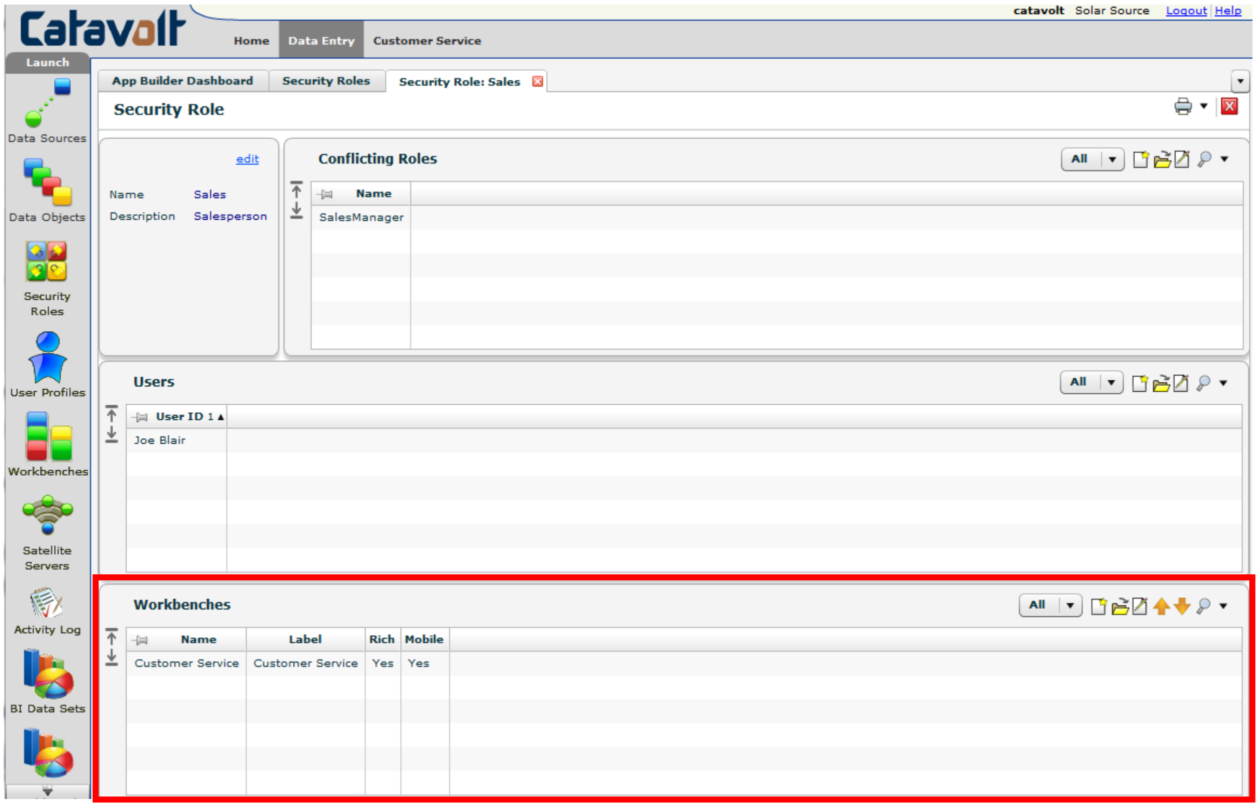
You can select a single or multiple User Profiles and press the Add button to add them to the Security Role.

After the User Security Role has been created, opening the selected record will display the details for the User Profile.

Security Role Workbenches

User Profiles that do not have Workbenches defined on the profile itself will automatically pick up Workbenches from the Security Roles that they are assigned to. You can alter the order the Workbenches appear by arranging the list of Security Roles on the User Profile as well as arranging the list of Workbenches on the Security Profile. Note that if a Workbench exists on multiple Security Roles, the first instance will be used (duplicate appearances will be removed).

Note that this feature is also available for users who have been authenticated remotely and have had Security Roles dynamically added during the authentication process. Remote authentication is beyond the scope of this document. Please contact Hexagon Support to request more information on this topic.



Security Role

edit

Name Sales

Description Salesperson

Conflicting Roles

Name
SalesManager

Users

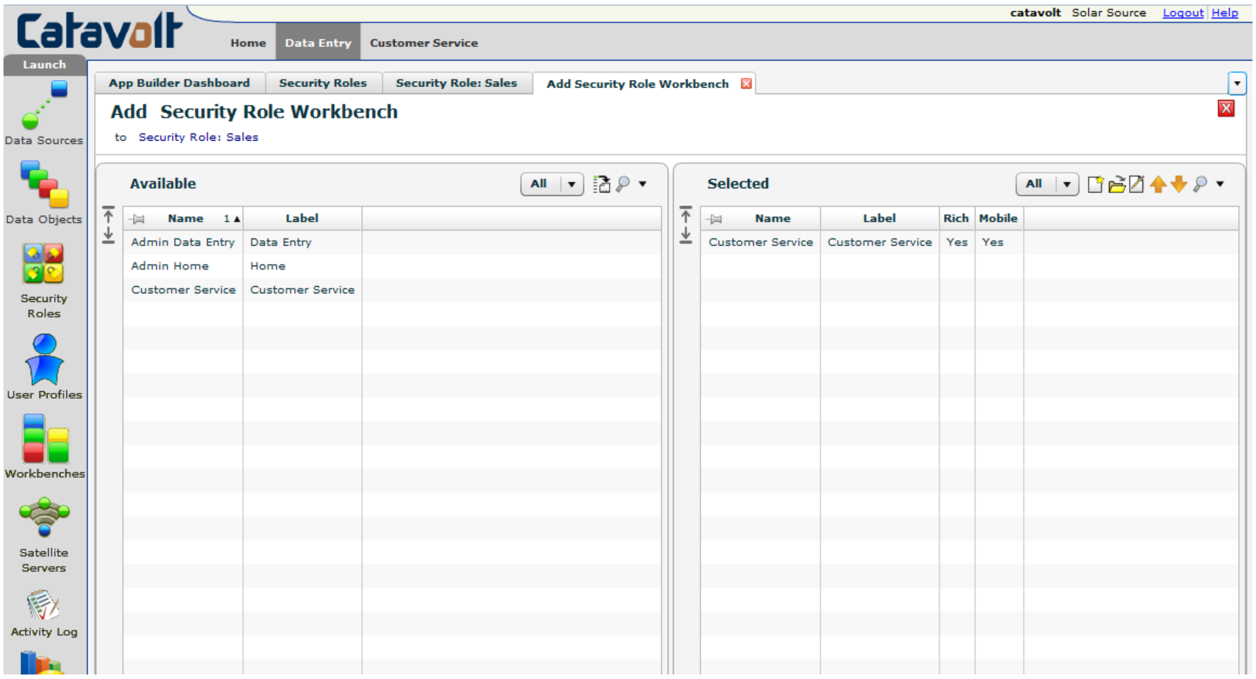
User ID 1 ▲
Joe Blair

Workbenches

Name	Label	Rich	Mobile
Customer Service	Customer Service	Yes	Yes

Figure 10: The Security Role details with the Workbenches section highlighted

When adding Workbench Security Roles, you will be presented with two lists. The Available Workbenches list shows all Workbenches. The Selected Workbenches list shows the Workbenches that are assigned to this Security Role.



Add Security Role Workbench

to Security Role: Sales

Available

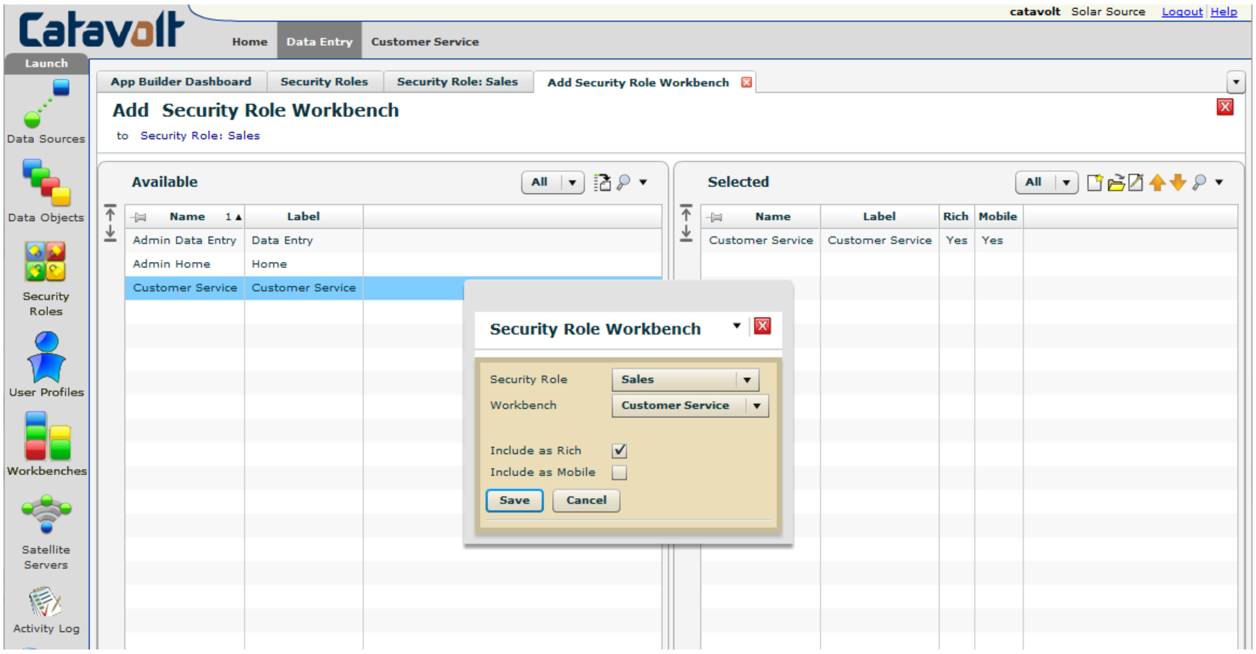
Name	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

Selected

Name	Label	Rich	Mobile
Customer Service	Customer Service	Yes	Yes

Figure 11: The add Security Role Workbench view.

You can select a single or multiple Workbenches and press the Add button to add them to the Security Role. If you select a single role, you will be presented with the following dialog:



Add Security Role Workbench

to Security Role: Sales

Available

Name	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

Selected

Name	Label	Rich	Mobile
Customer Service	Customer Service	Yes	Yes

Security Role Workbench

Security Role: Sales

Workbench: Customer Service

Include as Rich ☑

Include as Mobile □

Figure 12: The add Security Role Workbench dialog.

When creating a User Workbench, Security Role specifies the role that is to be granted access to the Workbench. Security Role displays a list of all roles in the system. The current Security Role you are working with will be the default selected value.

Workbench specifies the Workbench to which the Security Role is to be added. Workbench displays a list of all Workbenches in the system.

Include as Rich specifies whether the role should have access to this workbench when running the rich (web) client.

Include as Mobile specifies whether the role should have access to this workbench when running the mobile client.

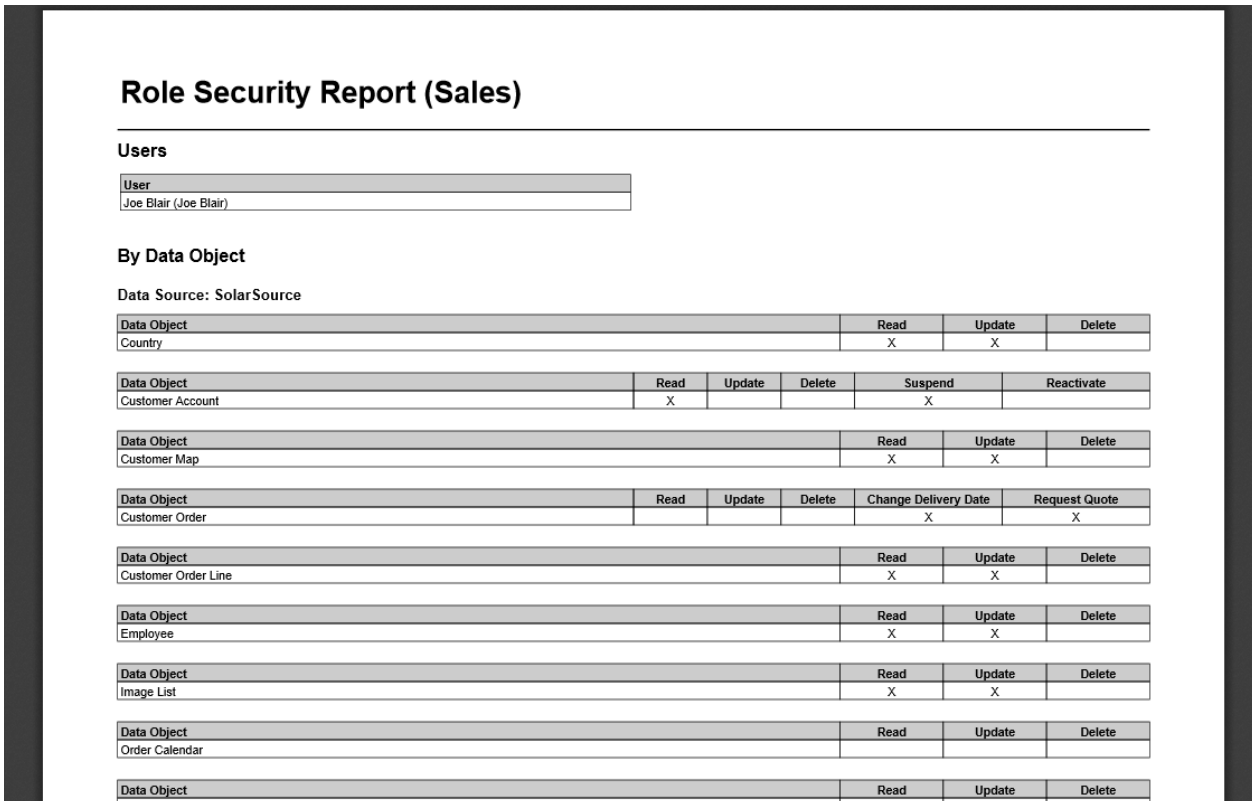
After the Security Role Workbench has been created, opening the Used By record will display the details for the Workbench.

Other Security Role Menu Actions

Print Security Report

Selecting this menu option will generate a PDF document that describes in detail the authorizations granted to this Security Role. It will provide a list of all User Profiles that are members of this Security Role. In addition, a list of all Data Objects and Actions are included along with their permissions.

The Role Security Report will look similar to the following picture:



Role Security Report (Sales)

Users

User
Joe Blair (Joe Blair)

By Data Object

Data Source: SolarSource

Data Object	Read	Update	Delete
Country	X	X	

Data Object	Read	Update	Delete	Suspend	Reactivate
Customer Account	X			X	

Data Object	Read	Update	Delete
Customer Map	X	X	

Data Object	Read	Update	Delete	Change Delivery Date	Request Quote
Customer Order				X	X

Data Object	Read	Update	Delete
Customer Order Line	X	X	

Data Object	Read	Update	Delete
Employee	X	X	

Data Object	Read	Update	Delete
Image List	X	X	

Data Object	Read	Update	Delete
Order Calendar			

Data Object	Read	Update	Delete

Figure 13: A sample of a generated Role Security Report

Types of Security

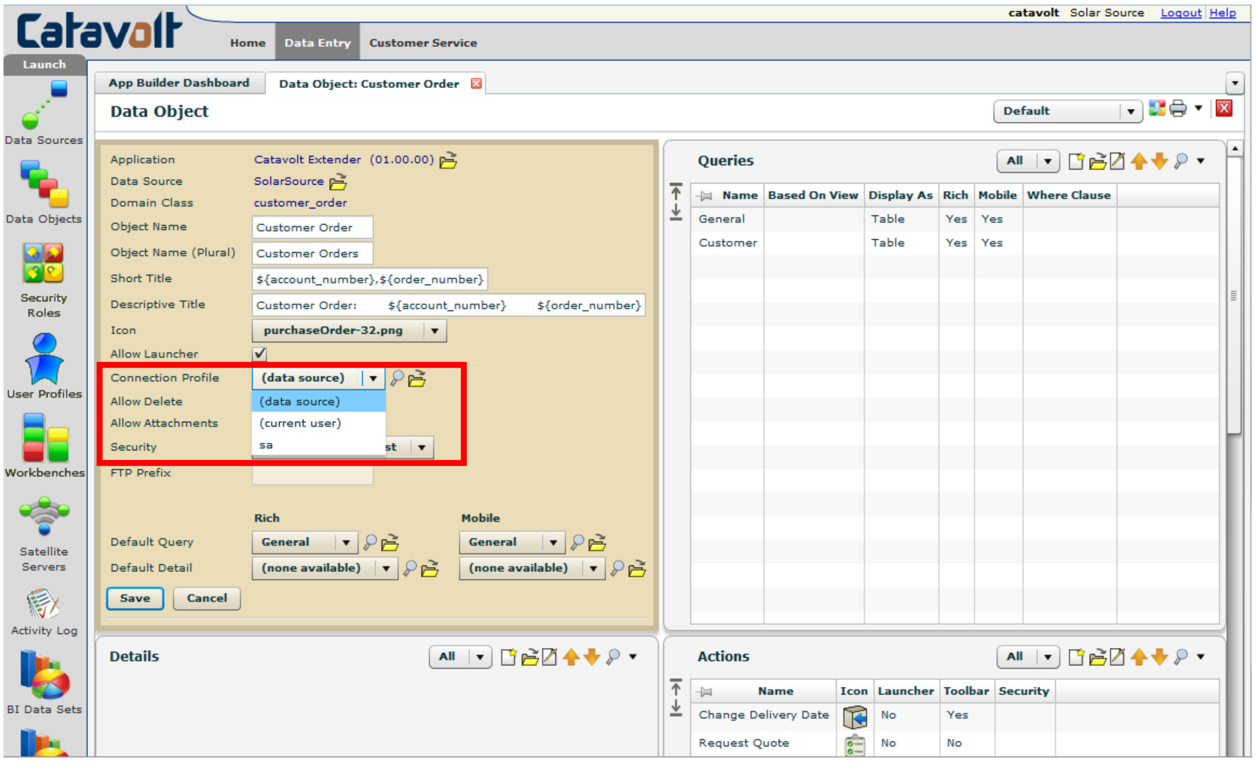
Xalt Runtime Security

Xalt runtime security is the method used to restrict Xalt users' access to data and operations. This is accomplished by assigning User Profiles to Security Roles as outlined above. Once users have been assigned to Security Roles, you can authorize Security Roles to Data Objects and Data Object Actions using Data Object Permissions and Action Permissions. See Chapter 3: Data Object

Permissions and Chapter 6: Data Object Action Permissions for more information on adding permissions to Data Objects and Data Object Actions.

Access to Back-End Data

As explained earlier, Xalt | Mobility accesses all back-end data via Data Sources. Only data that is accessible via the login credentials provided to the Data Source can be returned to Xalt. You have the ability to further refine this access for each Data Object using the Connection Profile option:



Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	customer_order
Object Name	Customer Order
Object Name (Plural)	Customer Orders
Short Title	${account_number},${order_number}
Descriptive Title	Customer Order: ${account_number} ${order_number}
Icon	purchaseOrder-32.png
Allow Launcher	☑
Connection Profile	(data source) (data source) (current user) sa
Allow Delete	
Allow Attachments	
Security	st
FTP Prefix	

	Rich	Mobile
Default Query	General	General
Default Detail	(none available)	(none available)

Save Cancel

Name	Based On View	Display As	Rich	Mobile	Where Clause
General		Table	Yes	Yes	
Customer		Table	Yes	Yes	

Name	Icon	Launcher	Toolbar	Security
Change Delivery Date		No	Yes	
Request Quote		No	No	

Figure 14: A Data Object detail with Connection Profile information highlighted

Each Data Object can choose to use the Data Source primary login credentials, an alternate set of credentials, or even pass through the user's login credentials to the back-end system when attempting to retrieve data. Only data that is accessible by the provided credentials will be returned.

Xalt Development Security

Xalt development security is the method used to restrict the ability to create and manipulate Xalt applications. By default, only User Profiles that are members of the Developer security role can maintain Xalt applications (Data Objects, Queries, Details, Actions, etc.). In addition, the Administrator security role has some limited capabilities, mostly in the areas of security and deployment (Data Sources, Workbenches, and Permissions).

You can further restrict Xalt Development Security by Connector gateway. By authorizing Security Roles to a Connector gateway, you are specifying that only these roles are allowed to create, manipulate, and deploy Xalt applications for the Connector gateway. This includes restricting maintenance on Data Objects, Data Sources, Workbenches, etc. that contain objects related to the specified Connector gateway. It is most often used in cases where you have multiple Connector gateways (e.g. one for Production and one for Development) and you need to restrict who can make changes to objects on the Production Connector gateway.

Other Ways of Limiting Data

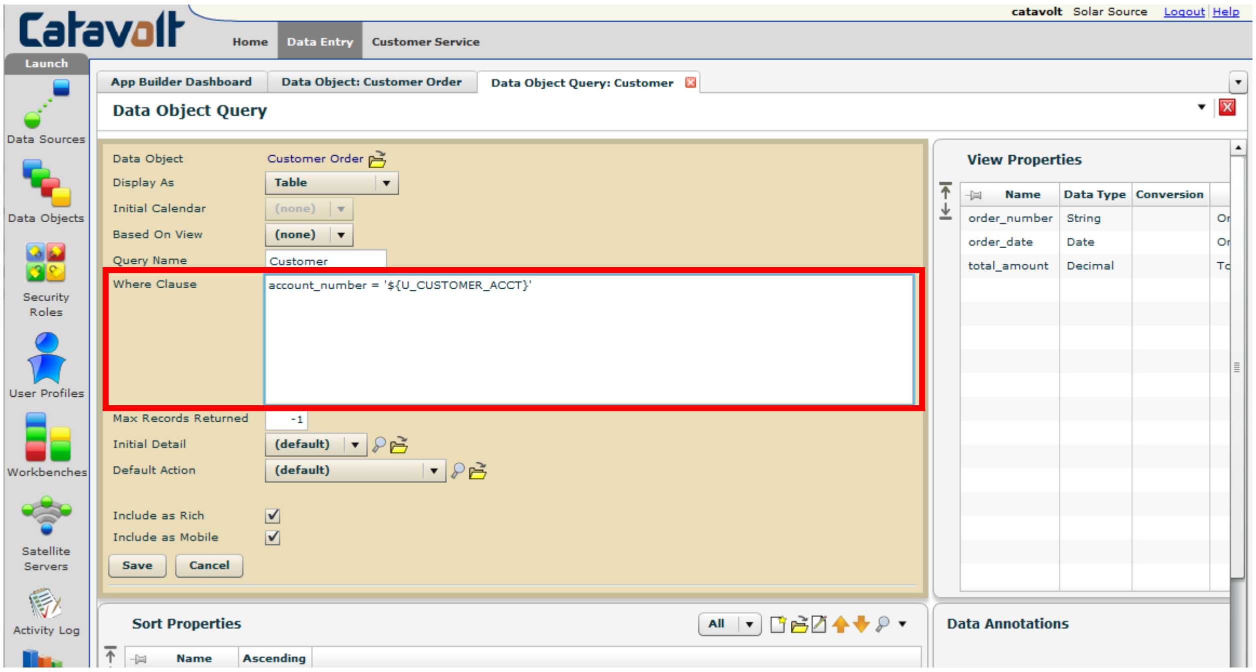
While technically not methods of security, you can also limit what data users can access via other means.

Workbenches

By assigning specific workbenches to a User Profile, you can limit which Data Objects they can access. Users can only access Data Objects from Workbenches they are assigned to, as well as any related objects that they can navigate to from these workbench launchers.

Where Clauses and User Properties

Lists of data (Queries, Detail Query Sections, etc.) usually contain Where Clauses that can be used to filter the list to only display certain records. By pairing this technique with User Properties, you can restrict individual users to seeing only certain records.

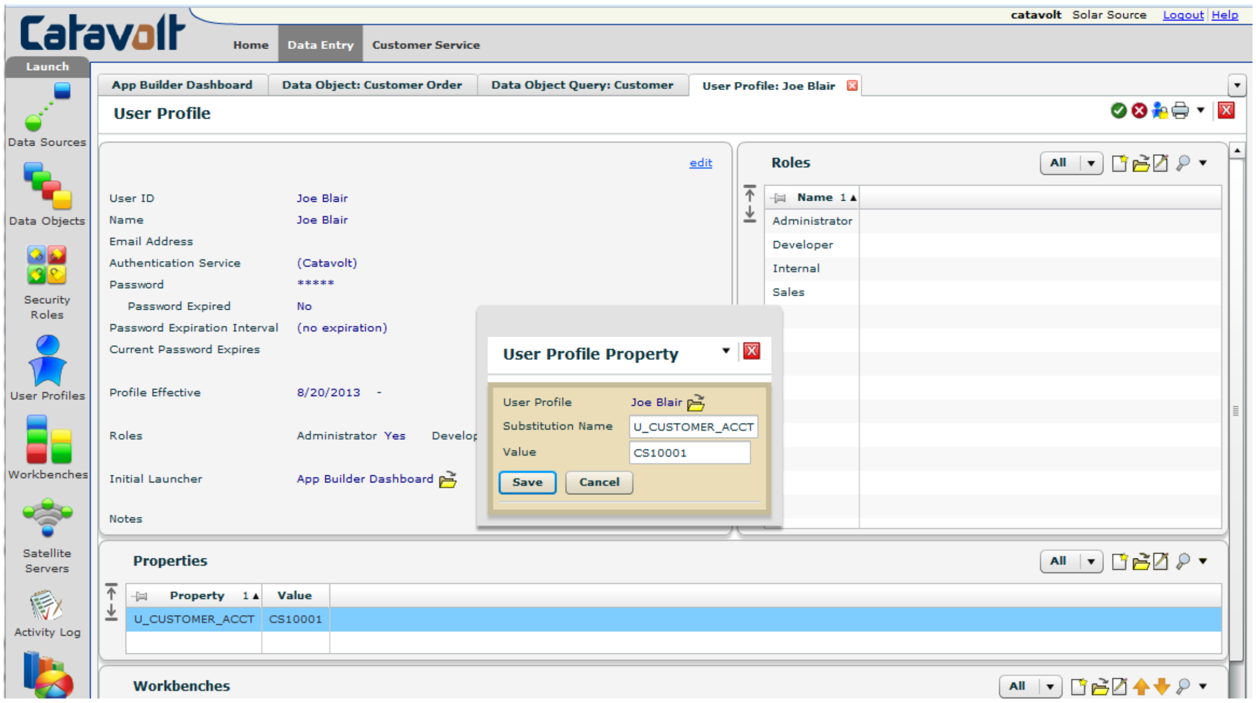


Data Object	Customer Order
Display As	Table
Initial Calendar	(none)
Based On View	(none)
Query Name	Customer
Where Clause	account_number = '${U_CUSTOMER_ACCT}'
Max Records Returned	-1
Initial Detail	(default)
Default Action	(default)
Include as Rich	☑
Include as Mobile	☑

Name	Data Type	Conversion
order_number	String	
order_date	Date	
total_amount	Decimal	

Figure 15: A Data Object Query detail with Where Clause highlighted

In the example above, the Customer Query has a where clause of account_number = '${U_CUSTOMER_ACCT}'



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Develop
Initial Launcher	App Builder Dashboard
Notes	

Roles

Name 1 ▴
Administrator
Developer
Internal
Sales

User Profile Property

User Profile	Joe Blair
Substitution Name	U_CUSTOMER_ACCT
Value	CS10001

Properties

Property 1 ▴	Value
U_CUSTOMER_ACCT	CS10001

Workbenches

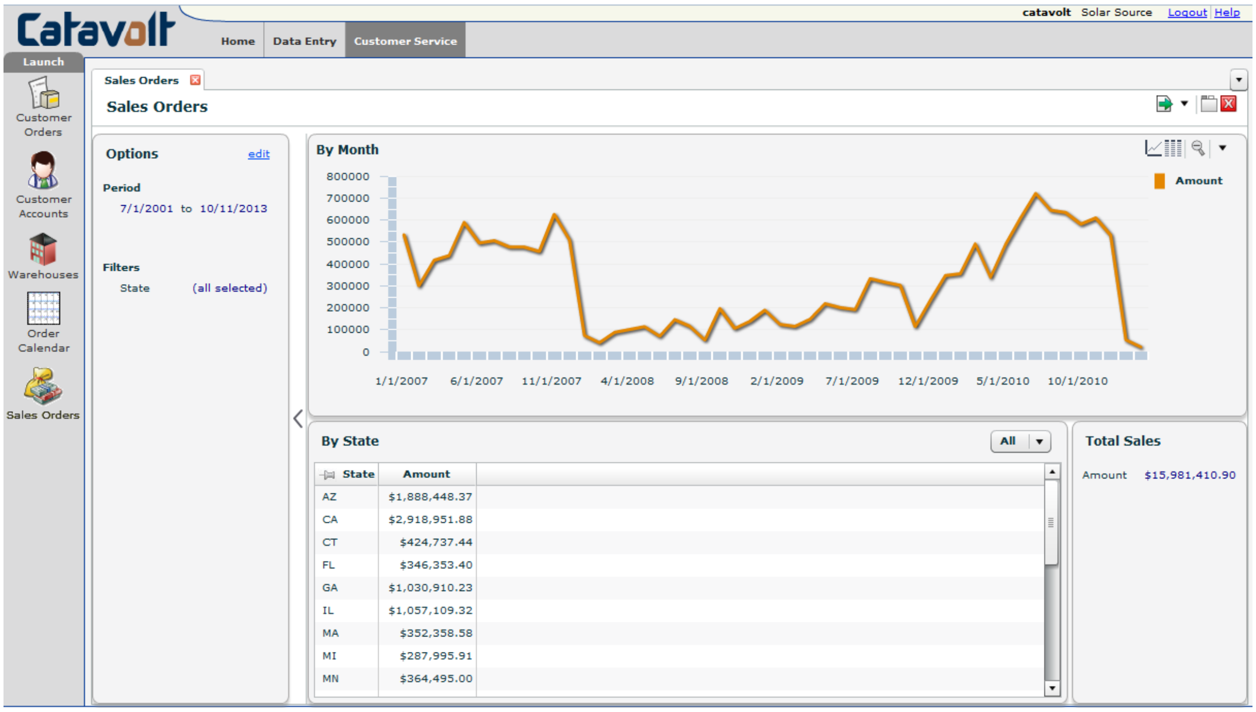
Figure 16: A User Profile Property create window for the U_CUSTOMER_ACCT property

In the User Profile, we are setting Joe Blair to have a U_CUSTOMER_ACCT value of CS01001. With this combination, Joe Blair will only be able to see Customer Orders for his Customer number (CS01001). Each User Profile can have a different U_CUSTOMER_ACCT value, which will allow all User Profiles to access the same Data Object (Customer Orders) while only seeing their own data.

Chapter 11: Business Intelligence (BI)

Chapter Summary

Xalt | Mobility has added the ability to create light Business Intelligence (BI) applications. Hexagon BI allows you to create data sets by connecting to existing SSAS data cubes as well as building in-memory data cubes from existing SQL databases using Mondrian. You can then create BI Dashboards that connect to these data sets to produce graphs, tables, and details from the dimensions and measurements in the data sets.



Sales Orders

Options edit

Period
7/1/2001 to 10/11/2013

Filters
State (all selected)

By State

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32
MA	$352,358.58
MI	$287,995.91
MN	$364,495.00

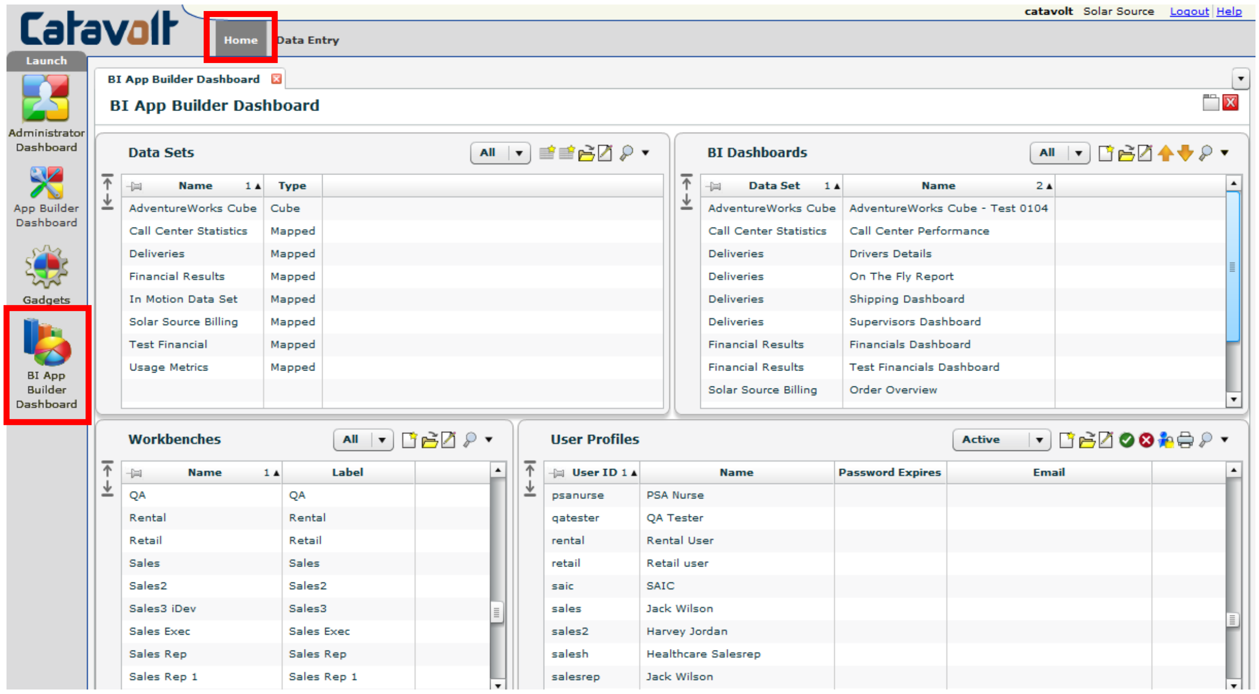
Total Sales

Amount $15,981,410.90

Figure 1: Example of a BI Dashboard with Graph, Table, and Detail sections

Accessing BI

To start creating BI applications, click on the BI App Builder Dashboard launcher on the Home workbench.



BI App Builder Dashboard

Data Sets

Name	Type
AdventureWorks Cube	Cube
Call Center Statistics	Mapped
Deliveries	Mapped
Financial Results	Mapped
In Motion Data Set	Mapped
Solar Source Billing	Mapped
Test Financial	Mapped
Usage Metrics	Mapped

BI Dashboards

Data Set	Name
AdventureWorks Cube	AdventureWorks Cube - Test 0104
Call Center Statistics	Call Center Performance
Deliveries	Drivers Details
Deliveries	On The Fly Report
Deliveries	Shipping Dashboard
Deliveries	Supervisors Dashboard
Financial Results	Financials Dashboard
Financial Results	Test Financials Dashboard
Solar Source Billing	Order Overview

Workbenches

Name	Label
QA	QA
Rental	Rental
Retail	Retail
Sales	Sales
Sales2	Sales2
Sales3 iDev	Sales3
Sales Exec	Sales Exec
Sales Rep	Sales Rep
Sales Rep 1	Sales Rep 1

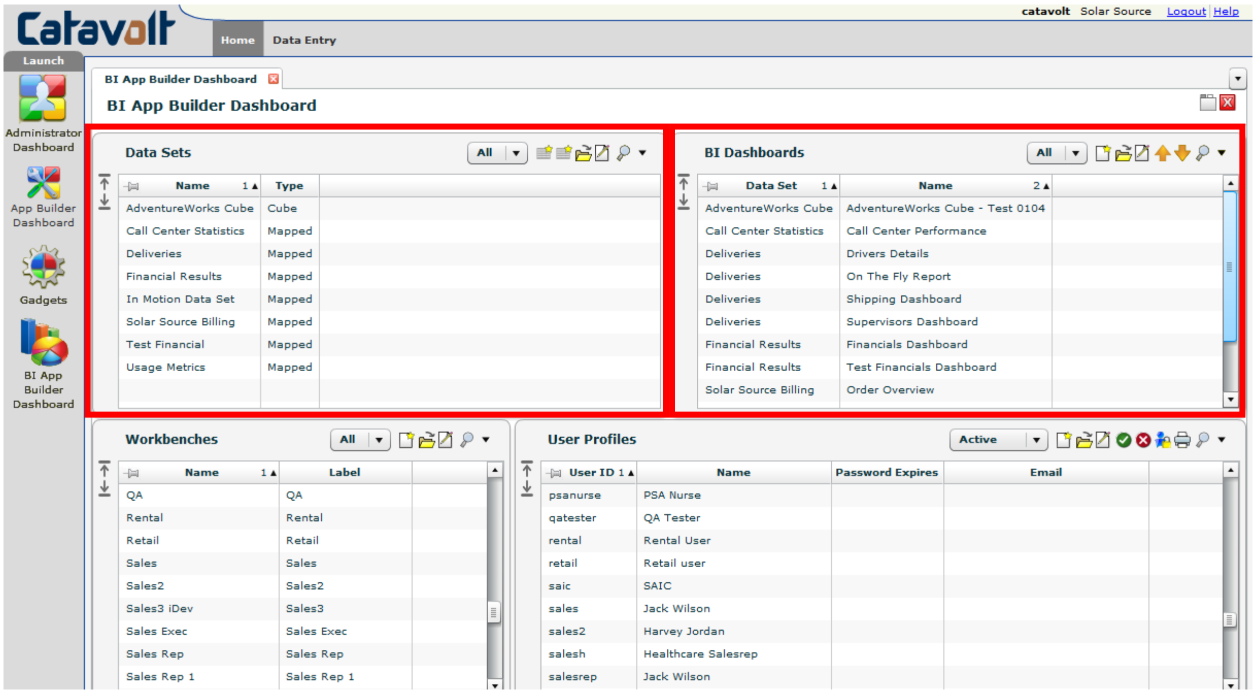
User Profiles

User ID	Name	Password Expires	Email
psanurse	PSA Nurse		
qatester	QA Tester		
rental	Rental User		
retail	Retail user		
saic	SAIC		
sales	Jack Wilson		
sales2	Harvey Jordan		
salesh	Healthcare Salesrep		
salesrep	Jack Wilson		

Figure 2: BI App Builder Dashboard details view with the Workbench and Launcher highlighted

BI Components

Hexagon BI is made up of 2 components: Data Sets (a list of Data Sets that can be used to retrieve data to show on BI Dashboards), and BI Dashboards (a list of Dashboards that can be launched from Workbenches).



BI App Builder Dashboard

Data Sets

Name 1 ▲	Type
AdventureWorks Cube	Cube
Call Center Statistics	Mapped
Deliveries	Mapped
Financial Results	Mapped
In Motion Data Set	Mapped
Solar Source Billing	Mapped
Test Financial	Mapped
Usage Metrics	Mapped

BI Dashboards

Data Set 1 ▲	Name 2 ▲
AdventureWorks Cube	AdventureWorks Cube - Test 0104
Call Center Statistics	Call Center Performance
Deliveries	Drivers Details
Deliveries	On The Fly Report
Deliveries	Shipping Dashboard
Deliveries	Supervisors Dashboard
Financial Results	Financials Dashboard
Financial Results	Test Financials Dashboard
Solar Source Billing	Order Overview

Workbenches

Name 1 ▲	Label
QA	QA
Rental	Rental
Retail	Retail
Sales	Sales
Sales2	Sales2
Sales3 iDev	Sales3
Sales Exec	Sales Exec
Sales Rep	Sales Rep
Sales Rep 1	Sales Rep 1

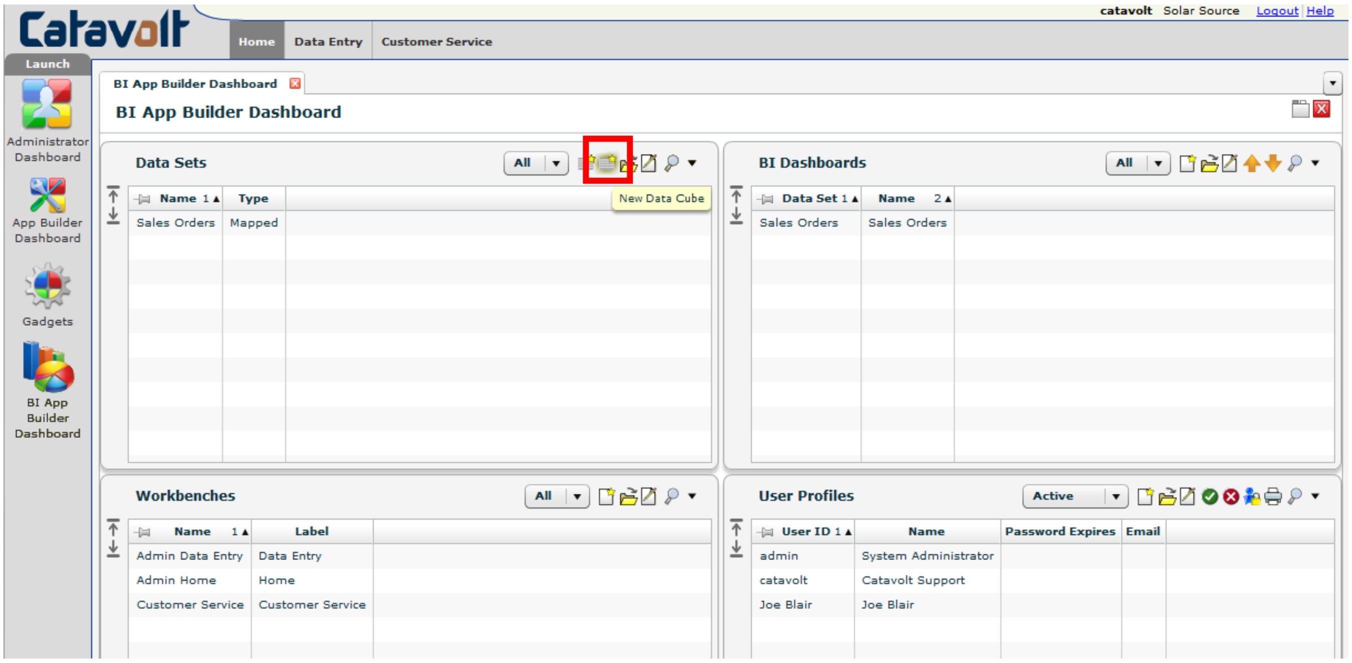
User Profiles

User ID 1 ▲	Name	Password Expires	Email
psanurse	PSA Nurse		
qatester	QA Tester		
rental	Rental User		
retail	Retail user		
saic	SAIC		
sales	Jack Wilson		
sales2	Harvey Jordan		
salesh	Healthcare Salesrep		
salesrep	Jack Wilson		

Figure 3: BI App Builder Dashboard with the Data Sets and BI Dashboards sections highlighted

Cube Data Sets

To create a new Cube Data Set, choose the New Data Cube menu option.



BI App Builder Dashboard

Data Sets

Name 1 ▲	Type
Sales Orders	Mapped

BI Dashboards

Data Set 1 ▲	Name 2 ▲
Sales Orders	Sales Orders

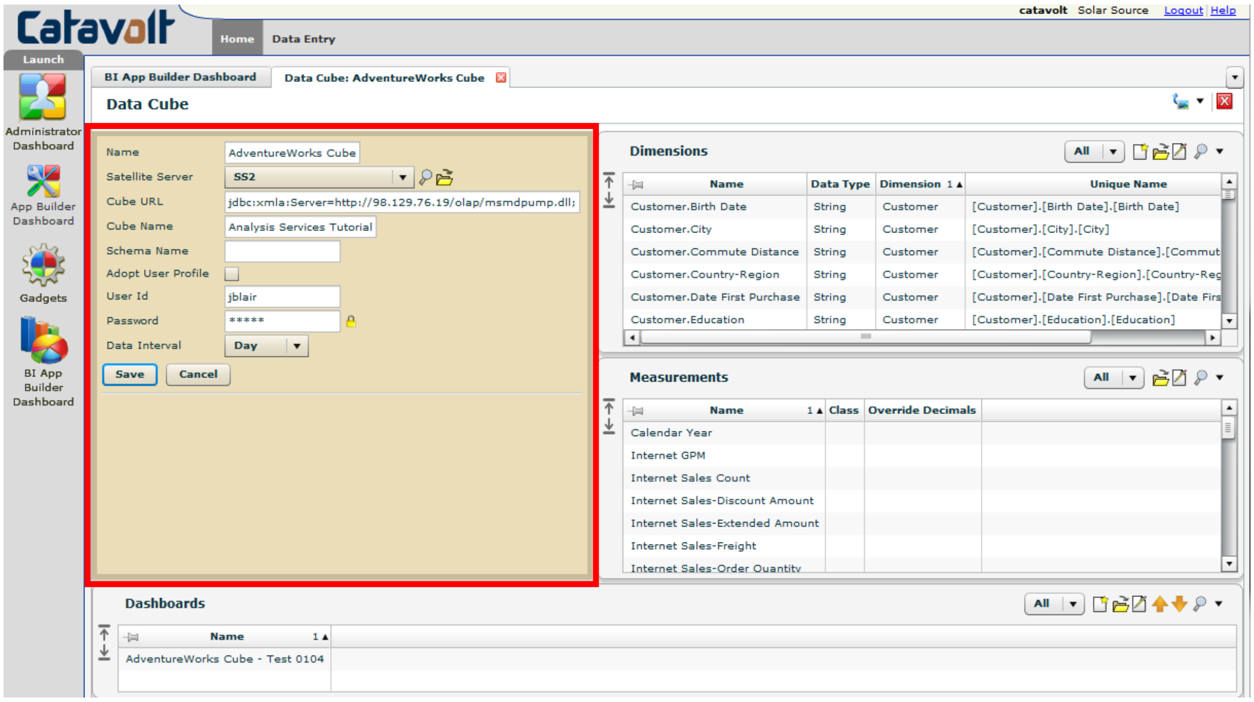
Workbenches

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Customer Service	Customer Service

User Profiles

User ID 1 ▲	Name	Password Expires	Email
admin	System Administrator		
catavolt	Catavolt Support		
Joe Blair	Joe Blair		

Figure 4: BI App Builder Dashboard section with the New Data Cube action highlighted



Data Cube

Name	AdventureWorks Cube
Satellite Server	SS2
Cube URL	jdbc:xmla:Server=http://98.129.76.19/olap/msmdpump.dll;
Cube Name	Analysis Services Tutorial
Schema Name	
Adopt User Profile	□
User Id	jblair
Password	*****
Data Interval	Day

Dimensions

Name	Data Type	Dimension 1 ▲	Unique Name
Customer.Birth Date	String	Customer	[Customer].[Birth Date].[Birth Date]
Customer.City	String	Customer	[Customer].[City].[City]
Customer.Commute Distance	String	Customer	[Customer].[Commute Distance].[Commut
Customer.Country-Region	String	Customer	[Customer].[Country-Region].[Country-Reg
Customer.Date First Purchase	String	Customer	[Customer].[Date First Purchase].[Date Firs
Customer.Education	String	Customer	[Customer].[Education].[Education]

Measurements

Name 1 ▲	Class	Override Decimals
Calendar Year		
Internet GPM		
Internet Sales Count		
Internet Sales-Discount Amount		
Internet Sales-Extended Amount		
Internet Sales-Freight		
Internet Sales-Order Quantity		

Dashboards

Name 1 ▲
AdventureWorks Cube - Test 0104

Figure 5: BI Cube Data Set details create screen

When creating a Data Cube, you must specify a Name. This value is used to uniquely identify the Data Set to the system and will not be displayed to the user.

Connector gateway specifies the Connector gateway that will be used to retrieve data from the back-end data cube.

Cube URL specifies the XMLA (XML for Analysis) URL that Xalt will use to connect to the back-end data cube. Xalt uses a standard olap4j (jdbc) driver to connect to the back-end data cube.

Cube Name specifies the name of the back-end data cube.

Schema Name specifies the schema name (if applicable) of the back-end data cube.

Adopt User Profile specifies which credentials should be passed to the back-end data cube when retrieving data. If this value is checked, the user id / password of the logged-in user will be passed to the back-end data cube. If this value is unchecked, the User Id and Password fields below will be used.

User Id specifies the User Id that should be passed to the back-end data cube when retrieving data. Note that this value will only be used if the Adopt User Profile value above is unchecked.

Password specifies the Password that should be passed to the back-end data cube when retrieving data. Note that this value will only be used if the Adopt User Profile value above is unchecked.

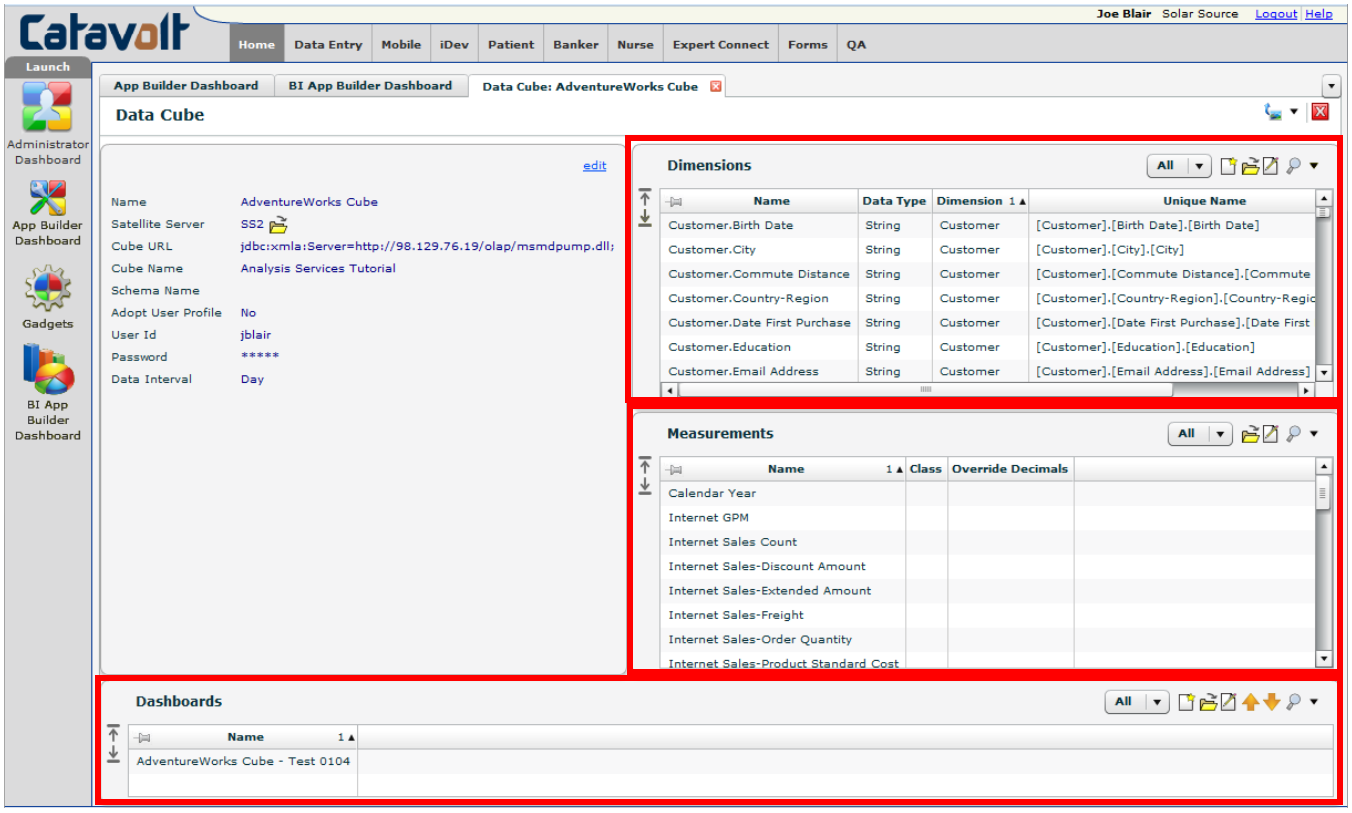
Data Interval specifies the date interval at which records are recorded in the cube. The available options are:

- (none) – There is no specific interval for dates in this data set.
- Day – Dates are recorded by day in this data set.
- Week – Dates are recorded by week in this data set.
- Month – Dates are recorded by month in this data set.

- Quarter – Dates are recorded by calendar quarter in this data set.
- Year – Dates are recorded by year in this data set.

Cube Data Set Components

Cube Data Sets are made up of 3 components: Dimensions (a list of Dimension properties for this Data Set), Measurements (a list of Measurement properties for this Data Set), and BI Dashboards (a list of Dashboards that access this Data Set).



Data Cube

Name	AdventureWorks Cube
Satellite Server	SS2
Cube URL	jdbc:xmla:Server=http://98.129.76.19/olap/msmdpump.dll;
Cube Name	Analysis Services Tutorial
Schema Name	
Adopt User Profile	No
User Id	jblair
Password	*****
Data Interval	Day

Dimensions

Name	Data Type	Dimension 1 ▲	Unique Name
Customer.Birth Date	String	Customer	[Customer].[Birth Date].[Birth Date]
Customer.City	String	Customer	[Customer].[City].[City]
Customer.Commute Distance	String	Customer	[Customer].[Commute Distance].[Commute
Customer.Country-Region	String	Customer	[Customer].[Country-Region].[Country-Regic
Customer.Date First Purchase	String	Customer	[Customer].[Date First Purchase].[Date First
Customer.Education	String	Customer	[Customer].[Education].[Education]
Customer.Email Address	String	Customer	[Customer].[Email Address].[Email Address]

Measurements

Name 1 ▲	Class	Override Decimals
Calendar Year		
Internet GPM		
Internet Sales Count		
Internet Sales-Discount Amount		
Internet Sales-Extended Amount		
Internet Sales-Freight		
Internet Sales-Order Quantity		
Internet Sales-Product Standard Cost		

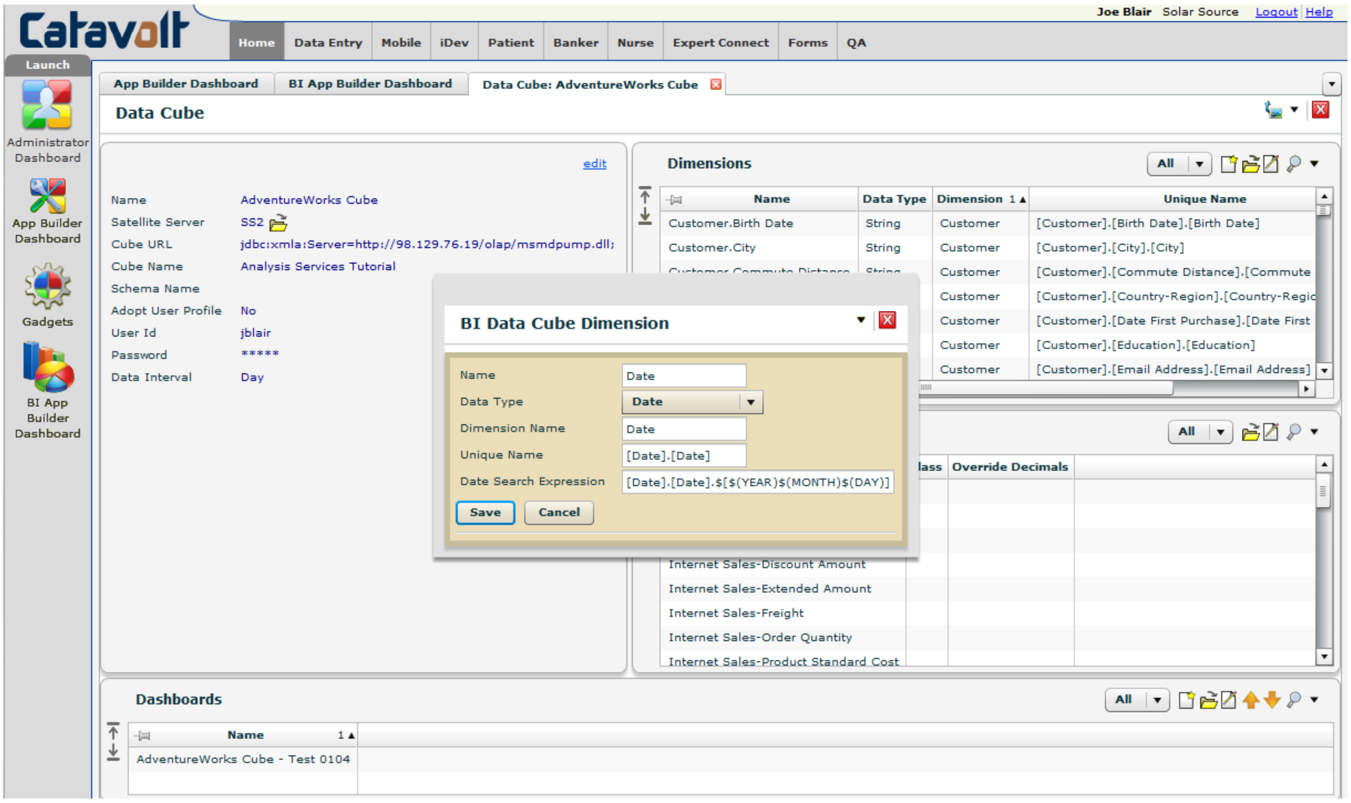
Dashboards

Name 1 ▲
AdventureWorks Cube - Test 0104

Figure 6: BI Cube Data Set details view with the Dimensions, Measurements, and Dashboards sections highlighted

Dimensions

When you create a Data Cube, all available Dimensions from the default hierarchy are automatically included in the Dimensions list. If you want to use dimensions from alternate hierarchies, you can also create a new Dimension by selecting the New menu option.



Data Cube

Name	AdventureWorks Cube
Satellite Server	SS2
Cube URL	jdbc:xmla:Server=http://98.129.76.19/olap/msmdpump.dll;
Cube Name	Analysis Services Tutorial
Schema Name	
Adopt User Profile	No
User Id	jblair
Password	*****
Data Interval	Day

Dimensions

Name	Data Type	Dimension 1 ▲	Unique Name
Customer.Birth Date	String	Customer	[Customer].[Birth Date].[Birth Date]
Customer.City	String	Customer	[Customer].[City].[City]
Customer.Commute Distance	String	Customer	[Customer].[Commute Distance].[Commute
		Customer	[Customer].[Country-Region].[Country-Regic
		Customer	[Customer].[Date First Purchase].[Date First
		Customer	[Customer].[Education].[Education]
		Customer	[Customer].[Email Address].[Email Address]

BI Data Cube Dimension

Name	Date
Data Type	Date
Dimension Name	Date
Unique Name	[Date].[Date]
Date Search Expression	[Date].[Date].$[$(YEAR)$(MONTH)$(DAY)]

	lass	Override Decimals
Internet Sales-Discount Amount		
Internet Sales-Extended Amount		
Internet Sales-Freight		
Internet Sales-Order Quantity		
Internet Sales-Product Standard Cost		

Dashboards

Name 1 ▲
AdventureWorks Cube - Test 0104

Figure 7: BI Data Cube Dimension create details view

When creating a Dimension, you must specify a Name. This value is used to uniquely identify the Dimension to the system and will not be directly displayed to the user.

Data Type specifies the data type of the Dimension. Available values are Date, Decimal, Whole Number, and String.

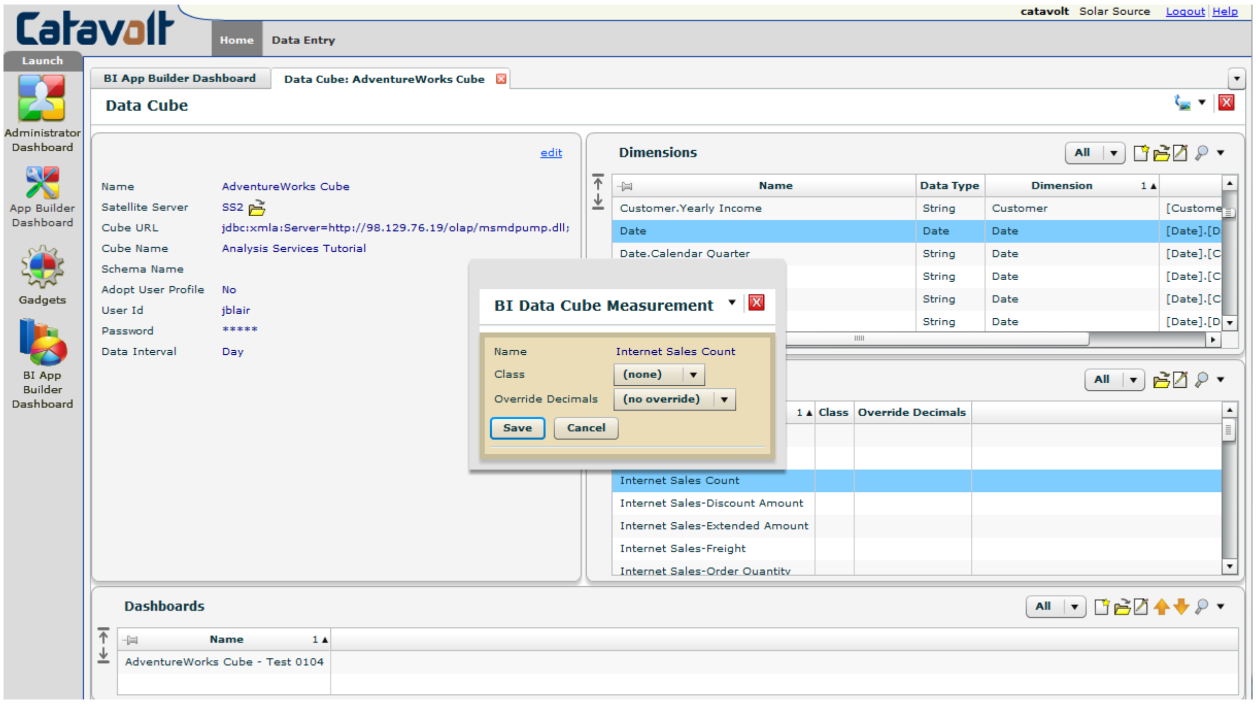
Dimension Name specifies the name of the dimension in the back-end data cube.

Unique Name specifies the actual name that will be used to query the dimension from the data cube.

Date Search Expression specifies the format in which dates are stored in the cube, as dates can be stored in multiple ways (D-M-Y, M-D-Y, Date+Time, etc). This is used by Xalt when specifying date ranges in cube queries. You can specify ${YEAR}, ${QUARTER}, ${MONTH}, ${WEEK} and ${DAY} substitution values when specifying the date expression,

Measurements

When you create a Data Cube, all available Measurements are automatically included in the Dimensions list. You can edit a Measurement to change some of the behavior from the back-end data cube.



Data Cube

Name	AdventureWorks Cube
Satellite Server	SS2
Cube URL	jdbc:xmla:Server=http://98.129.76.19/olap/msmdpump.dll;
Cube Name	Analysis Services Tutorial
Schema Name	
Adopt User Profile	No
User Id	jblair
Password	*****
Data Interval	Day

Dimensions

Name	Data Type	Dimension 1 ▲	
Customer.Yearly Income	String	Customer	[Custome
Date	Date	Date	[Date].[D
Date.Calendar Quarter	String	Date	[Date].[C
	String	Date	[Date].[C
	String	Date	[Date].[C
	String	Date	[Date].[D

BI Data Cube Measurement

Name	Internet Sales Count
Class	(none)
Override Decimals	(no override)

1 ▲	Class	Override Decimals
Internet Sales Count		
Internet Sales-Discount Amount		
Internet Sales-Extended Amount		
Internet Sales-Freight		
Internet Sales-Order Quantity		

Dashboards

Name 1 ▲
AdventureWorks Cube - Test 0104

Figure 8: BI Data Cube Measurement create view

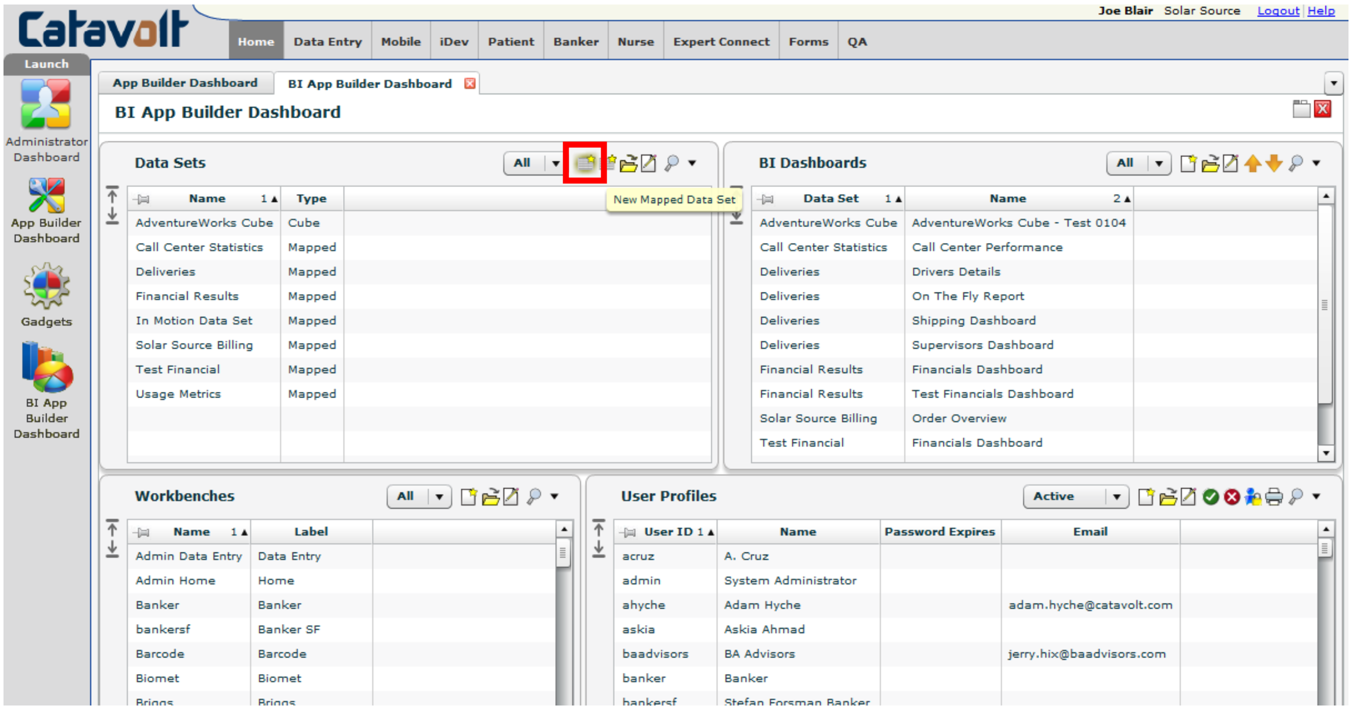
Class specifies special formatting options used to display values for this property. The following options are available:

- **(none)** – There is special formatting for this measurement.
- **Money** – Displayed as a monetary value with a $.
- **Percent** – Displayed as a percent value with percent sign. The value should be stored in the database in place value (e.g. .063). Xalt will automatically multiply and render the value as a percent (e.g. 6.3%).

Override Decimals specifies the number of decimals to display this measurement with. Allowable options are (no override) as well as overriding values to show any value between 0 and 5 decimals. Values will be rounded using the "half up" method.

Mapped Data Sets

To create a new Mapped Data Set, choose the New Mapped Data Set menu option.



BI App Builder Dashboard

Data Sets

Name 1 ▲	Type
AdventureWorks Cube	Cube
Call Center Statistics	Mapped
Deliveries	Mapped
Financial Results	Mapped
In Motion Data Set	Mapped
Solar Source Billing	Mapped
Test Financial	Mapped
Usage Metrics	Mapped

BI Dashboards

Data Set 1 ▲	Name 2 ▲
AdventureWorks Cube	AdventureWorks Cube - Test 0104
Call Center Statistics	Call Center Performance
Deliveries	Drivers Details
Deliveries	On The Fly Report
Deliveries	Shipping Dashboard
Deliveries	Supervisors Dashboard
Financial Results	Financials Dashboard
Financial Results	Test Financials Dashboard
Solar Source Billing	Order Overview
Test Financial	Financials Dashboard

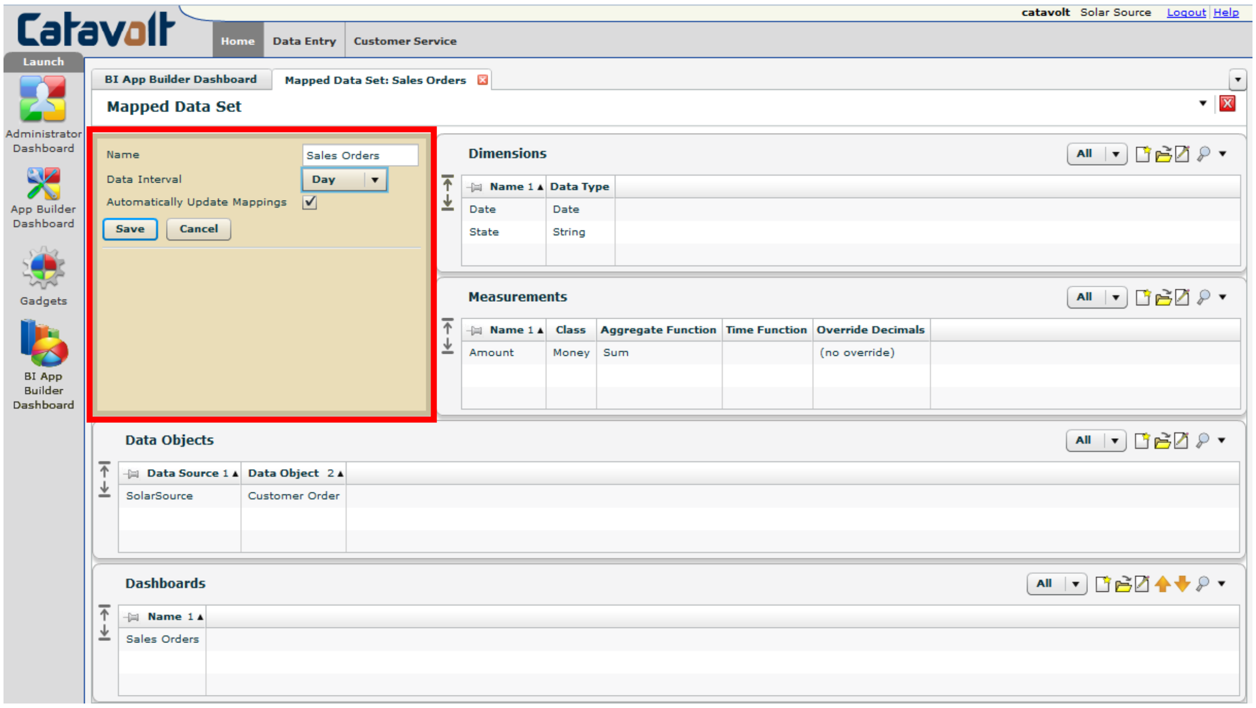
Workbenches

Name 1 ▲	Label
Admin Data Entry	Data Entry
Admin Home	Home
Banker	Banker
bankersf	Banker SF
Barcode	Barcode
Biomet	Biomet
Briggs	Briggs

User Profiles

User ID 1 ▲	Name	Password Expires	Email
acruz	A. Cruz		
admin	System Administrator		
ahyche	Adam Hyche		adam.hyche@catavolt.com
askia	Askia Ahmad		
baadvisors	BA Advisors		jerry.hix@baadvisors.com
banker	Banker		
bankersf	Stefan Forsman Banker		

Figure 9: BI App Builder Dashboard details view with the New Mapped Data Set action highlighted



Mapped Data Set

Name	Sales Orders
Data Interval	Day
Automatically Update Mappings	☑

Dimensions

Name 1 ▲	Data Type
Date	Date
State	String

Measurements

Name 1 ▲	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)

Data Objects

Data Source 1 ▲	Data Object 2 ▲
SolarSource	Customer Order

Dashboards

Name 1 ▲
Sales Orders

Figure 10: BI Mapped Data Set create details view

When creating a Mapped Data Set, you must specify a Name. This value is used to uniquely identify the Workbench to the system and will not be displayed to the user.

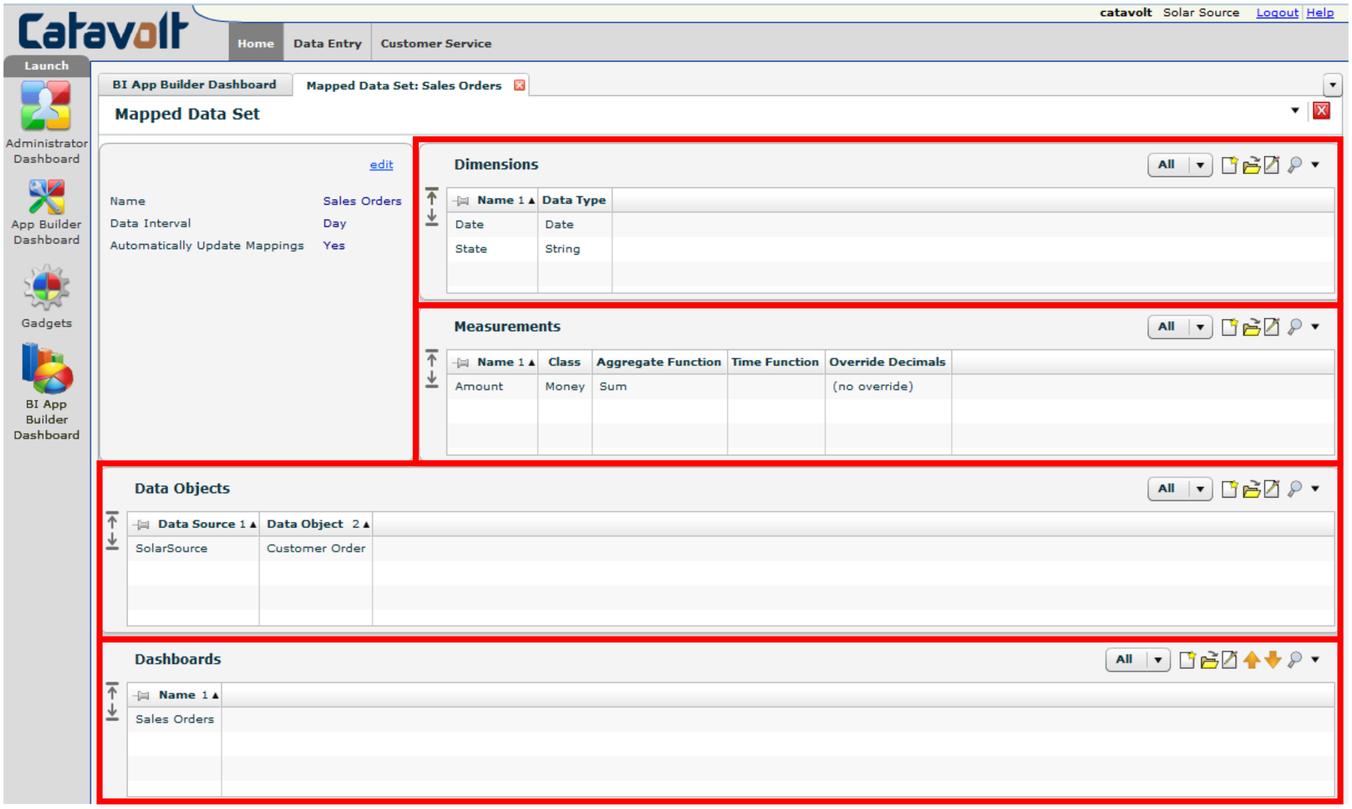
Data Interval specifies the date interval at which records are recorded in the data set. The available options are:

- (none) – There is no specific interval for dates in this data set.
- Day – Dates are recorded by day in this data set.
- Week – Dates are recorded by week in this data set.
- Month – Dates are recorded by month in this data set.
- Quarter – Dates are recorded by calendar quarter in this data set.
- Year – Dates are recorded by year in this data set.

Automatically Update Mappings specifies whether the Mondrian mapping files should be automatically updated as changes are made to this Data Set. If this option is unchecked, you will be responsible for manually updating the Mondrian mapping files as you make changes to the data set.

Mapped Data Set Components

Mapped Data Sets are made up of 4 components: Dimensions (a list of Dimension properties for this Data Set), Measurements (a list of Measurement properties for this Data Set), Data Objects (a list of Xalt Data Objects that provide the data behind the data set), and BI Dashboards (a list of Dashboards that access this Data Set).



Mapped Data Set

Name	Sales Orders
Data Interval	Day
Automatically Update Mappings	Yes

Dimensions

Name 1 ▲	Data Type
Date	Date
State	String

Measurements

Name 1 ▲	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)

Data Objects

Data Source 1 ▲	Data Object 2 ▲
SolarSource	Customer Order

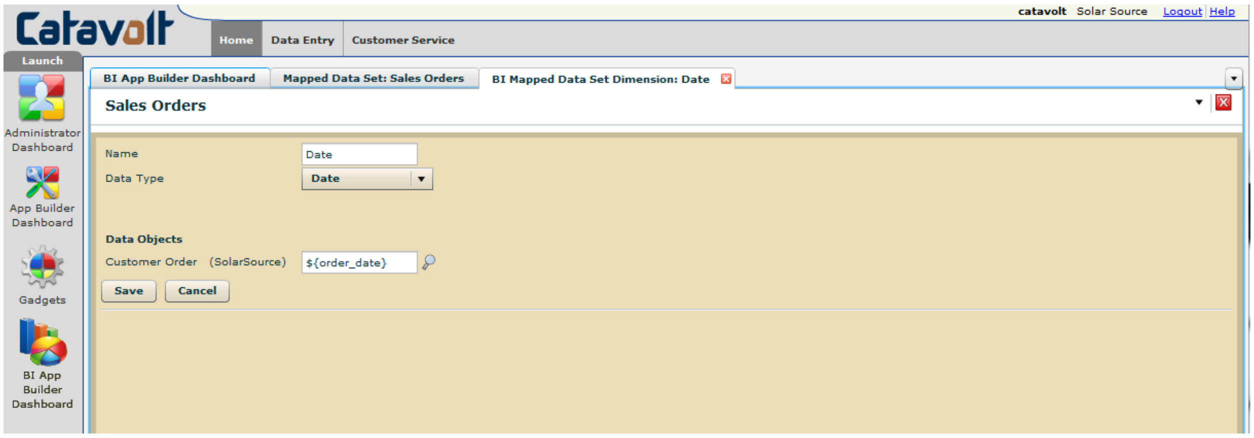
Dashboards

Name 1 ▲
Sales Orders

Figure 11: BI Mapped Data Set details view with the Dimensions, Measurements, Data Objects, and Dashboards sections highlighted

Dimensions

When you create a Mapped Data Set, you can create Dimensions to specify values that should be used to aggregate data (e.g. by Date, by State, by Gender, etc.).



Sales Orders

Name	Date
Data Type	Date

Data Objects

Customer Order (SolarSource)	${order_date}

Figure 12: BI Mapped Data Set Dimension create details view

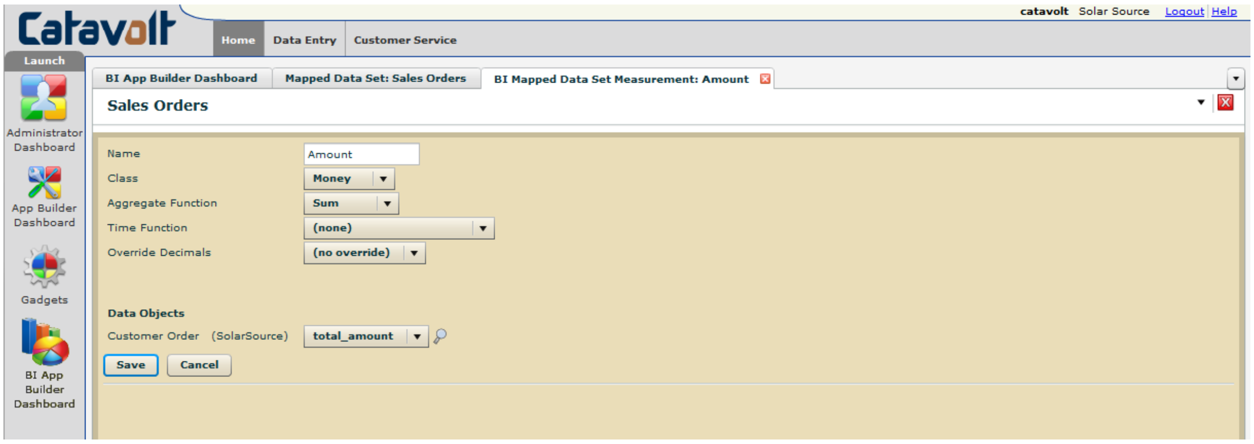
When creating a Dimension, you must specify a Name. This value is used to uniquely identify the Dimension to the system and will not be directly displayed to the user.

Data Type specifies the data type of the Dimension. Available values are Date, Decimal, Whole Number, and String.

Data Objects specifies which property in the Xalt Data Object maps to this Dimension. There will be one line for each Data Object defined in the Data Objects section. You may specify a property in the object's Additional Query Scope via a Substitution Value (or use the find button), you may specify a constant value, or you may leave the value blank to specify that the dimension is not mapped in this Data Object. See the Data Objects section in this chapter for more information about mapping values to dimensions.

Measurements

When you create a Mapped Data Set, you can create Measurements to specify decimal values that should be used to quantify data (e.g. Total Sales, Days Late, etc.).



Sales Orders

Name: Amount

Class: Money

Aggregate Function: Sum

Time Function: (none)

Override Decimals: (no override)

Data Objects

Customer Order (SolarSource): total_amount

Figure 13: BI Mapped Data Set Measurement create details view

When creating a Measurement, you must specify a Name. This value is used to uniquely identify the Dimension to the system and will not be directly displayed to the user.

Class specifies special formatting options used to display values for this property. The following options are available:

- **(none)** – There is special formatting for this measurement.
- **Money** – Displayed as a monetary value with a $.
- **Percent** – Displayed as a percent value with percent sign. The value should be stored in the database in place value (e.g. .063). Xalt will automatically multiply and render the value as a percent (e.g. 6.3%).

Aggregate Function specifies how the data set should aggregate/combine multiple records when creating slices from the data set:

- **Average** – Take the average value from the data records.
- **Count** – Take the count of the data records.
- **Max** – Take the max value from the data records.
- **Min** – Take the min value from the data records.
- **Sum** – Take the sum of value from the data records. This is the default.

Time Function specifies the time period and/or time function to retrieve values for. The following options are available:

- **(none)** – Take the value from the current time period.
- **Last Year Value** – Take the value from last year.
- **Last Year Difference** – Take the difference of the current value and last year's value.
- **Last Year Percent Difference** – Take the percent difference of the current value and last year's value.
- **Last Quarter Value** – Take the value from last calendar quarter.

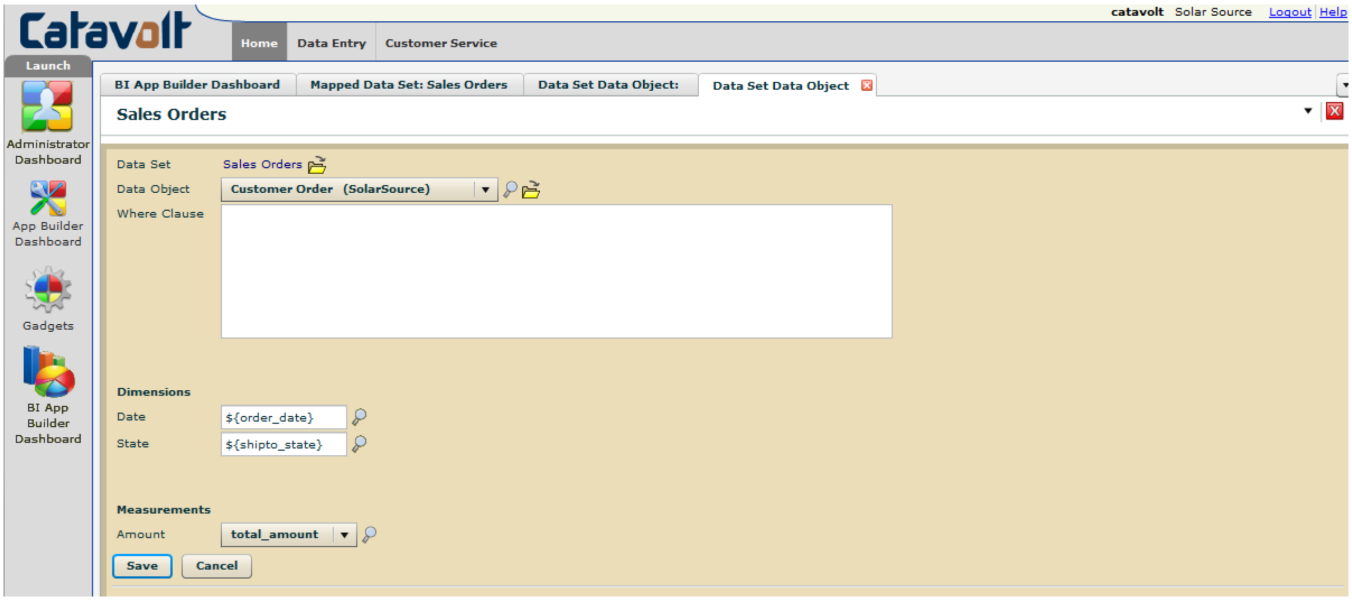
- Last Quarter Difference – Take the difference of the current value and last calendar quarter's value.
- Last Quarter Percent Difference – Take the percent difference of the current value and last calendar quarter's value.
- Last Month Value – Take the value from last month.
- Last Month Difference – Take the difference of the current value and last month's value.
- Last Month Percent Difference – Take the percent difference of the current value and last month's value.
- Last Week Value – Take the value from last week.
- Last Week Difference – Take the difference of the current value and last week's value.
- Last Week Percent Difference – Take the percent difference of the current value and last week's value.

Override Decimals specifies the number of decimals to display this measurement with. Allowable options are (no override) as well as overriding values to show any value between 0 and 5 decimals. Values will be rounded using the "half up" method.

Data Objects specifies which property in an Xalt Data Object maps to this Measurement. There will be one line for each Data Object defined in the Data Objects section. You may specify a property in the object's Additional Query Scope via a Substitution Value (or use the find button), you may specify a constant value, or you may leave the value blank to specify that the measurement is not mapped in this Data Object. See the Data Objects section in this chapter for more information about mapping values to measurements.

Data Objects

When you create a Mapped Data Set, you must specify one or more Data Objects. Data Objects refer to existing Xalt Data Objects (see Chapter 3: Data Objects for more information about Data Objects). Data Set Dimensions and Measurements map to Data Object properties and constants. These Data Objects are read and their data aggregated by Mondrian to build the Data Set.



Sales Orders

Data Set Sales Orders
Data Object Customer Order (SolarSource)
Where Clause

Dimensions
Date ${order_date}
State ${shipto_state}

Measurements
Amount total_amount

Figure 14: BI Mapped Data Set Domain Object create details view

When creating a Data Set Data Object, you must specify a Data Object. This value is an existing Xalt Data Object that is to be queried to build the Data Set.

Where Clause allows you to specify a Mondrian-supported where clause to restrict the records in the data set. This value is optional and can be left blank.

Dimensions specifies which properties in the Data Object map to each defined Dimension. There will be one line for each Dimension defined in the Dimensions section. You may specify a property in the object's Additional Query Scope via a Substitution Value (or use the find button), you may specify a constant value, or you may leave the value blank to specify that the dimension is not mapped in this Data Object.

Measurements specifies which properties in the Data Object map to each defined Measurement. There will be one line for each Measurement defined in the Measurement section. You may choose from a list of the Data Object's numeric properties, or you may specify (not mapped) to indicate that the measurement is not mapped in this Data Object.

Note that you may specify multiple Data Objects in this section. Data Objects are not required to be in the same Data Source, nor are Data Objects required to map every dimension and measurement. In addition, two Data Objects can map the same measurements and dimensions. This allows for different combinations when aggregating data on a single data set, for example:

1. You want to create a Sales Order data set, but only current data is in the Sales Order table. Older data is in a separate Sales Order History table. You can create two Data Objects, one for each table that maps all dimensions and measurements. Xalt will combine both sets of records into a single data set.
2. You want to create a Costs vs. Revenue data set, but cost information is in one table, while revenues are in another. You can create two Data Objects, each one mapping their own measurements (costs or revenue) over the same dimensions (date, line of business, etc.). Xalt will build records by combining both sets of data

View Data Set Mapping

The View Data Set Mapping menu option is available on Data Object to show you the Mondrian mapping XML that exists for the Data Object. An example of a Mapping XML is:

```
<?xml version="1.0" encoding="UTF-8"?>
<Schema name="TestSchema">
 <Cube name="TestCube" defaultMeasure="Amount">
  <Table name="customer_order" schema="sa" />
  <Dimension name="Date"  TYPE="TimeDimension"  foreignKey="order_date" >
   <Hierarchy hasAll="true" allMemberName="All Date"  primaryKey="datedimension" >
    <Table name="Hexagontimedimension" schema="sa" />
    <Level name="Year" uniqueMembers="false" column="year"  type="Numeric" levelType="TimeYears"
table="Hexagontimedimension" />
    <Level name="Quarter" uniqueMembers="false" column="quarter"  type="Numeric" levelType="TimeQuarters"
table="Hexagontimedimension" />
```

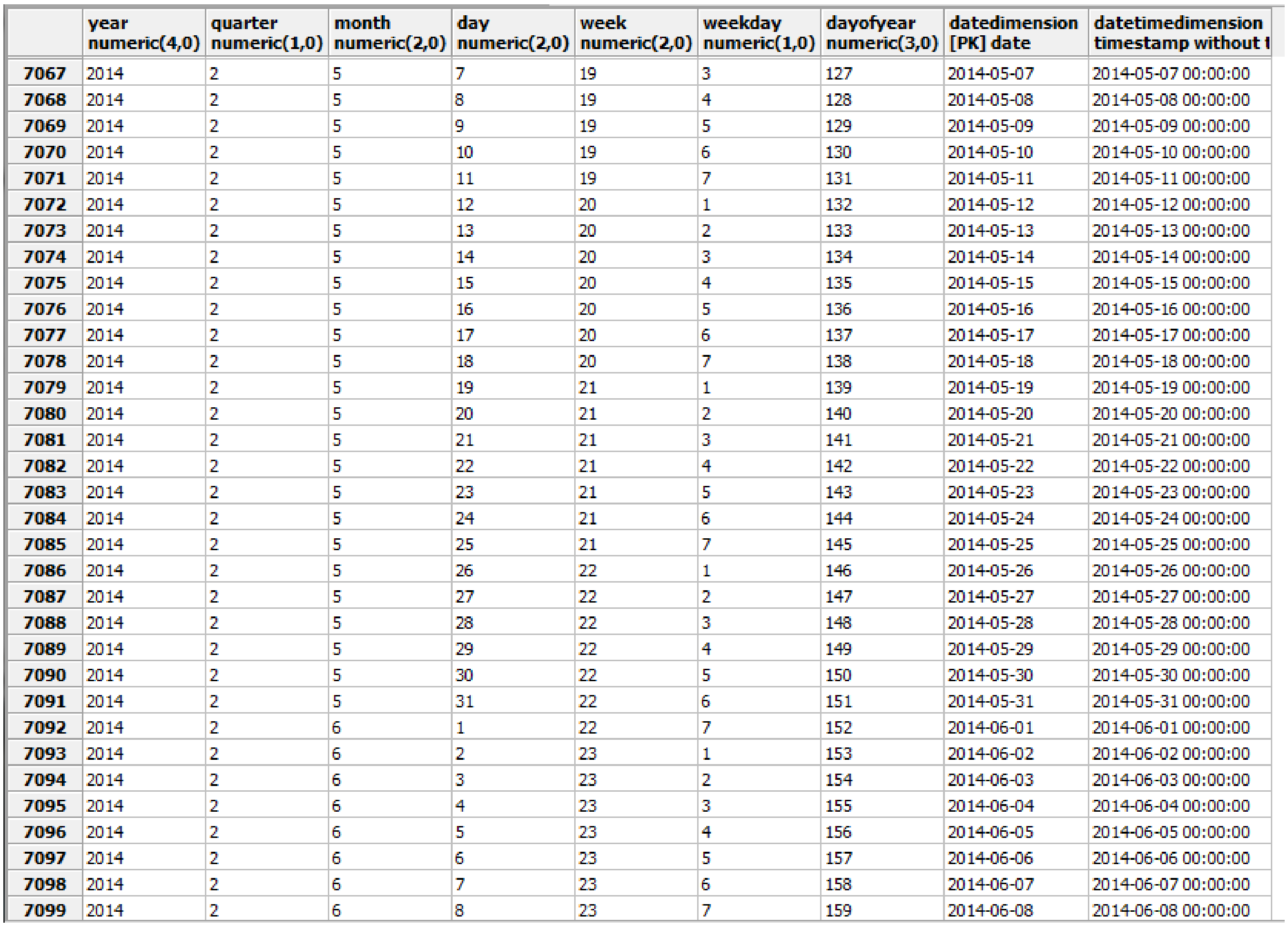
```
    <Level name="Month" uniqueMembers="false" column="month"  type="Numeric" levelType="TimeMonths"
table="Hexagontimedimension" />
    <Level name="Day" uniqueMembers="false" column="day"  type="Numeric" levelType="TimeDays"
table="Hexagontimedimension" />
    <Level name="Date" uniqueMembers="false" column="datedimension"  type="Date" levelType="TimeUndefined"
table="Hexagontimedimension" />
   </Hierarchy>
  </Dimension>
  <Dimension name="State"  foreignKey="shipto_state" >
   <Hierarchy hasAll="true" allMemberName="All State"  primaryKey="shipto_state" >
    <Level name="State" uniqueMembers="false"  column="shipto_state"  type="String" table="customer_order">
    </Level>
   </Hierarchy>
  </Dimension>
  <Measure name="Amount" column="total_amount"  aggregator="sum" formatString="Standard">
  </Measure>
 </Cube>
</Schema>
```

Note that the Mapping file is only updated if you select the Automatically Update Mappings option on the Data Set. For performance reasons, these mappings are not updated every time you make changes to the data set. Instead, when you make one or more changes, the mapping file will be updated the next time you launch a BI Dashboard over the Data Set.

Other Menu Options

Create BI Date Dimension Table

In order to be able to efficiently aggregate data based on dates, Mondrian needs information about how a particular date can be grouped (what week, month, year, etc. this date falls on). Xalt | Mobility has a Create BI Date Dimension Table menu option that will create a database table (Hexagontimedimension) in the back-end database that can be joined with other tables to provide this information. Below is an example of the table format.



	year numeric(4,0)	quarter numeric(1,0)	month numeric(2,0)	day numeric(2,0)	week numeric(2,0)	weekday numeric(1,0)	dayofyear numeric(3,0)	datedimension [PK] date	datetimedimension timestamp without t
7067	2014	2	5	7	19	3	127	2014-05-07	2014-05-07 00:00:00
7068	2014	2	5	8	19	4	128	2014-05-08	2014-05-08 00:00:00
7069	2014	2	5	9	19	5	129	2014-05-09	2014-05-09 00:00:00
7070	2014	2	5	10	19	6	130	2014-05-10	2014-05-10 00:00:00
7071	2014	2	5	11	19	7	131	2014-05-11	2014-05-11 00:00:00
7072	2014	2	5	12	20	1	132	2014-05-12	2014-05-12 00:00:00
7073	2014	2	5	13	20	2	133	2014-05-13	2014-05-13 00:00:00
7074	2014	2	5	14	20	3	134	2014-05-14	2014-05-14 00:00:00
7075	2014	2	5	15	20	4	135	2014-05-15	2014-05-15 00:00:00
7076	2014	2	5	16	20	5	136	2014-05-16	2014-05-16 00:00:00
7077	2014	2	5	17	20	6	137	2014-05-17	2014-05-17 00:00:00
7078	2014	2	5	18	20	7	138	2014-05-18	2014-05-18 00:00:00
7079	2014	2	5	19	21	1	139	2014-05-19	2014-05-19 00:00:00
7080	2014	2	5	20	21	2	140	2014-05-20	2014-05-20 00:00:00
7081	2014	2	5	21	21	3	141	2014-05-21	2014-05-21 00:00:00
7082	2014	2	5	22	21	4	142	2014-05-22	2014-05-22 00:00:00
7083	2014	2	5	23	21	5	143	2014-05-23	2014-05-23 00:00:00
7084	2014	2	5	24	21	6	144	2014-05-24	2014-05-24 00:00:00
7085	2014	2	5	25	21	7	145	2014-05-25	2014-05-25 00:00:00
7086	2014	2	5	26	22	1	146	2014-05-26	2014-05-26 00:00:00
7087	2014	2	5	27	22	2	147	2014-05-27	2014-05-27 00:00:00
7088	2014	2	5	28	22	3	148	2014-05-28	2014-05-28 00:00:00
7089	2014	2	5	29	22	4	149	2014-05-29	2014-05-29 00:00:00
7090	2014	2	5	30	22	5	150	2014-05-30	2014-05-30 00:00:00
7091	2014	2	5	31	22	6	151	2014-05-31	2014-05-31 00:00:00
7092	2014	2	6	1	22	7	152	2014-06-01	2014-06-01 00:00:00
7093	2014	2	6	2	23	1	153	2014-06-02	2014-06-02 00:00:00
7094	2014	2	6	3	23	2	154	2014-06-03	2014-06-03 00:00:00
7095	2014	2	6	4	23	3	155	2014-06-04	2014-06-04 00:00:00
7096	2014	2	6	5	23	4	156	2014-06-05	2014-06-05 00:00:00
7097	2014	2	6	6	23	5	157	2014-06-06	2014-06-06 00:00:00
7098	2014	2	6	7	23	6	158	2014-06-07	2014-06-07 00:00:00
7099	2014	2	6	8	23	7	159	2014-06-08	2014-06-08 00:00:00

Figure 15: Sample content of the Hexagontimedimension database table

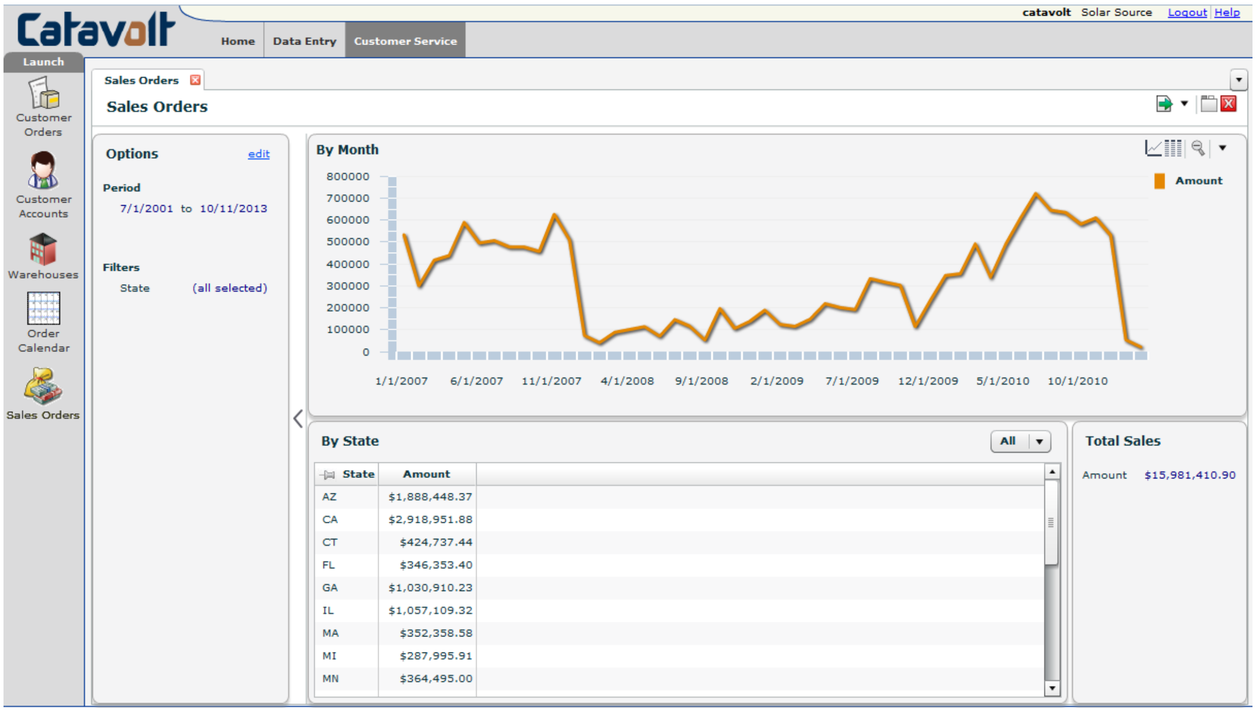
This table must be created before you can open BI Dashboards on the data set. The data source's primary userid must have authority to create tables in order to be able to perform this menu action.

Update Data Set Mapping

Mondrian uses an XML Mapping file to map dimensions and measurements to SQL tables. For performance reasons, these mappings are not updated every time you make changes to the data set. Instead, when you make one or more changes, the mapping file will be updated the next time you launch a BI Dashboard over the Data Set. You can force the Mapping files to update by running the Update Date Set Mapping menu option.

BI Dashboards

A BI Dashboard takes the data defined in Data Sets and renders them as graphs, tables, and details.



Period

7/1/2001 to 10/11/2013

Filters

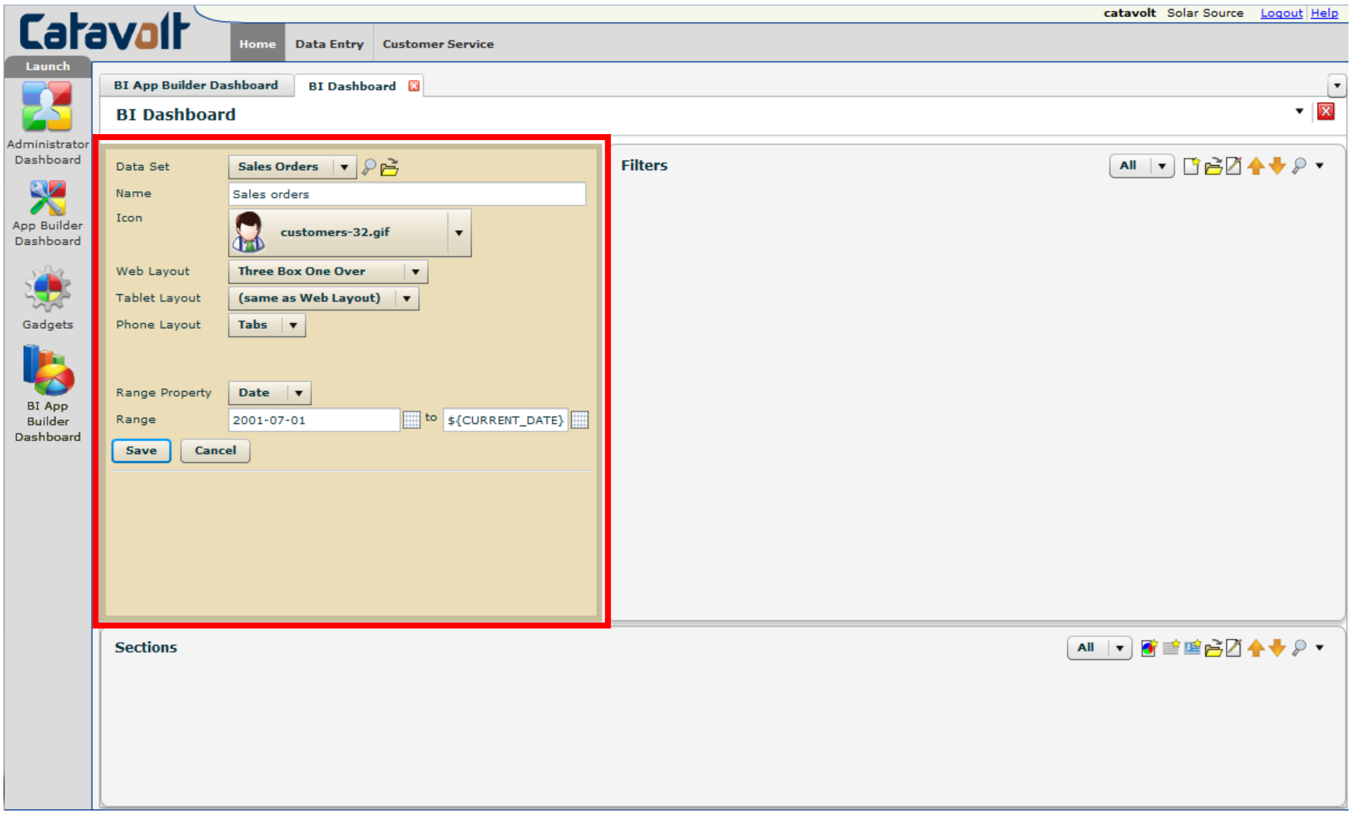
State (all selected)

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32
MA	$352,358.58
MI	$287,995.91
MN	$364,495.00

Amount $15,981,410.90

Figure 16: A BI Dashboard with Graph, Table, and Detail sections

Creating BI Dashboards



Data Set: Sales Orders
Name: Sales orders
Icon: customers-32.gif
Web Layout: Three Box One Over
Tablet Layout: (same as Web Layout)
Phone Layout: Tabs
Range Property: Date
Range: 2001-07-01 to ${CURRENT_DATE}

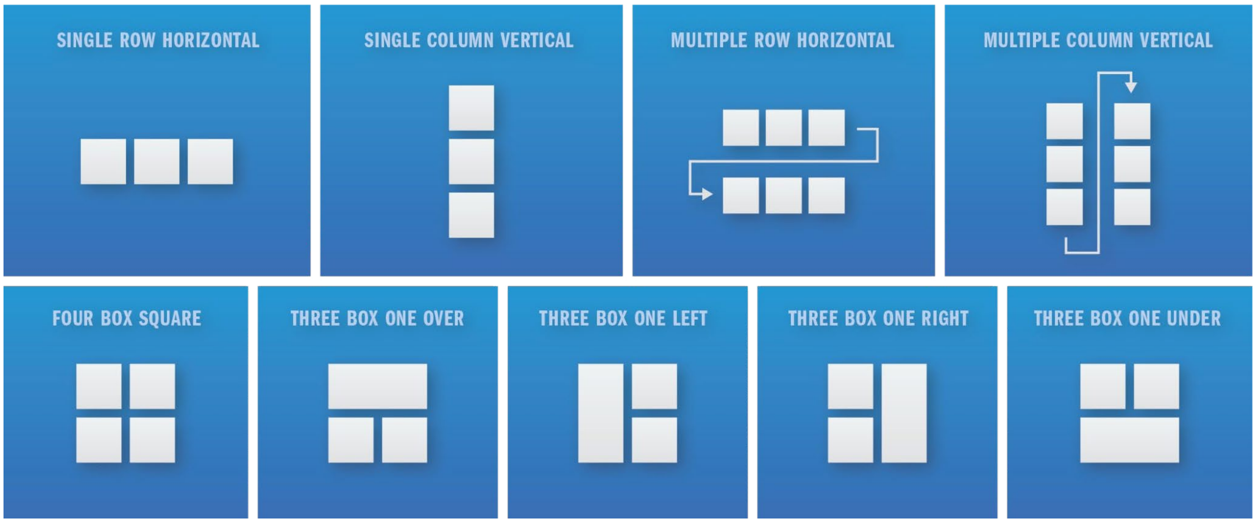
Figure 17: BI Dashboard create details view

When creating a BI Dashboard, you must specify a Data Set. This specifies the data set used to retrieve and display data.

Name specifies the name of the BI Workbench. Name will be used to default the label of the launcher for the BI Dashboard.

Icon allows you to specify the default image to display when adding this BI Dashboard to a Workbench. You can choose the default Hexagon image or any other image you have uploaded. See Chapter 2: Data Sources for more information on how to upload images.

Web Layout specifies how you want to arrange the different sections when viewed by a Web Browser. The following options are available:



- Single Row Horizontal – All sections are arrange side-by-side in a single row
- Single Column Vertical – All sections are arrange on top of each other in a single column
- Multiple Row Horizontal – Sections are arranged side-by-side. Sections fill up a row based on the window size and then wrap to the next row as needed. As the window is resized the sections adjust and wrap. This is similar to word wrapping in word processing applications.
- Multiple Column Vertical – Sections are arranged on top of each other. Sections fill up a column based on the window size and then wrap to the top of the next column. As the window is resized the sections adjust and wrap.
- Four Box Square – Up to 4 sections are equally sized and displayed in a 2x2 area. Any sections beyond the fourth are added underneath using the "Single Column Vertical" style.
- Three Box One Over – Up to 3 sections are placed, the first on row one and the other two side-by-side on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Left – Up to 3 sections are placed, the first on column one and the other two on top of each other on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Right – Up to 3 sections are placed, the first two on top of each other on column one and the third on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Under – Up to 3 sections are placed, the first two side-by-side on row one and the third on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Tabs – Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

Tablet Layout specifies how you want to arrange the different sections when viewed on a Tablet-size mobile device. The following options are available:

- (same as web layout) – Use the same layout as the rich client, if it is available on mobile. If the web layout is not available on the mobile device, Single Column Vertical will be used.
- Single Column Vertical – All sections are arrange on top of each other in a single column
- Four Box Square – Up to 4 sections are equally sized and displayed in a 2x2 area. Any sections beyond the fourth are added underneath using the "Single Column Vertical" style.
- Three Box One Over – Up to 3 sections are placed, the first on row one and the other two side-by-side on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.

- Three Box One Left – Up to 3 sections are placed, the first on column one and the other two on top of each other on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Right – Up to 3 sections are placed, the first two on top of each other on column one and the third on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Under – Up to 3 sections are placed, the first two side-by-side on row one and the third on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Tabs - Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

Phone Layout specifies how you want to arrange the different sections when viewed on a Phone-size mobile device. The following options are available:

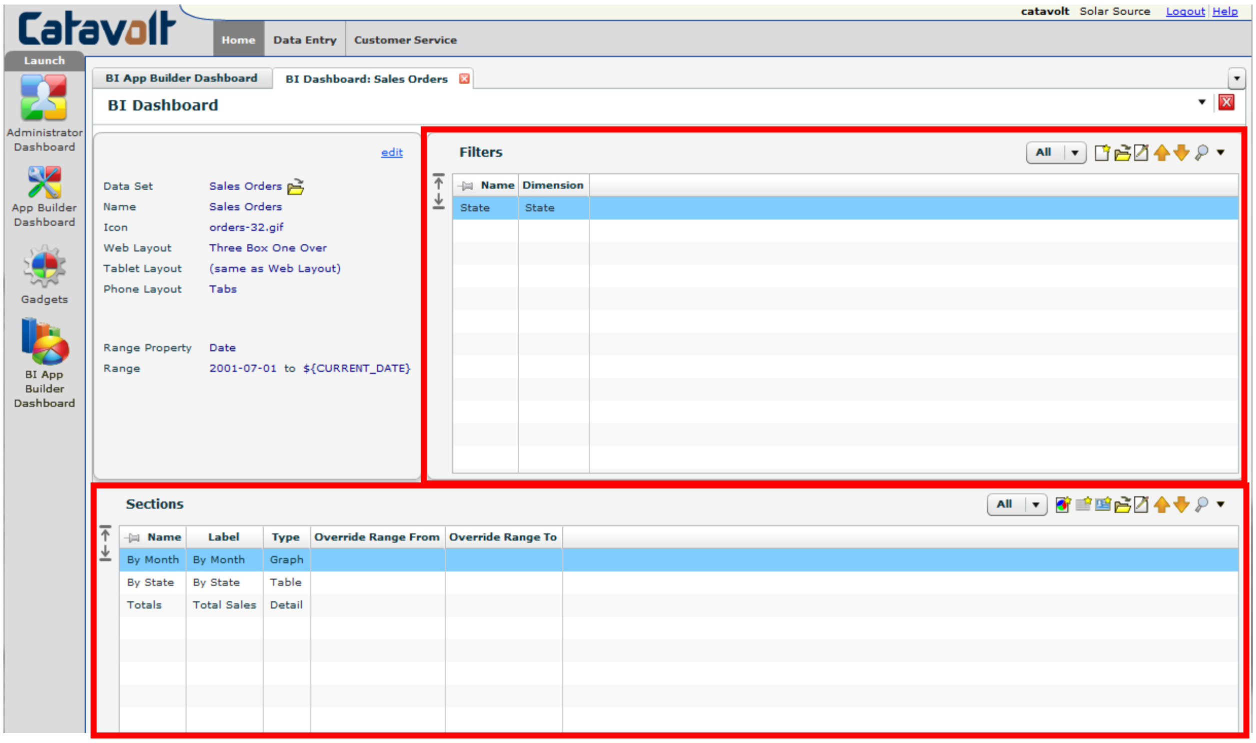
- Tabs - Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

Range Property specifies the data set dimension that will define the range over which this BI Dashboard is reporting against. This is typically a Date property

Range specifies the from- and to- values that will constrain the range over which this BI Dashboard is reporting against.

BI Dashboard Components

BI Dashboards are made up of 2 components: Filters (a list of Dimension properties that can be used to filter results) and Sections (a list of Graphs, Tables and Details to present data from the data set).



BI Dashboard

Data Set	Sales Orders
Name	Sales Orders
Icon	orders-32.gif
Web Layout	Three Box One Over
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
Range Property	Date
Range	2001-07-01 to ${CURRENT_DATE}

Filters

Name	Dimension
State	State

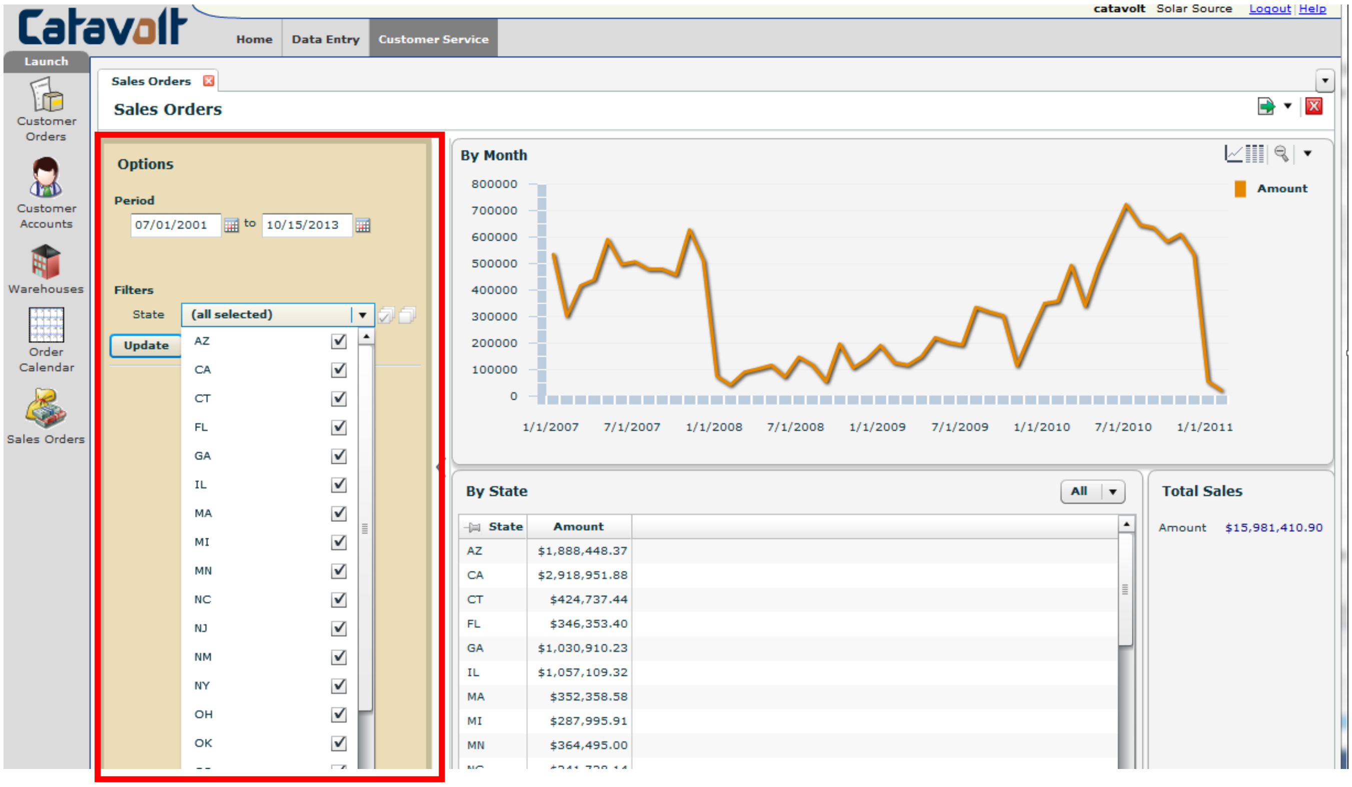
Sections

Name	Label	Type	Override Range From	Override Range To
By Month	By Month	Graph		
By State	By State	Table		
Totals	Total Sales	Detail		

Figure 18: BI Dashboard details view with Filters and Sections sections highlighted

BI Dashboard Filters

BI Dashboard Filters allow you to add dimensions as dynamic filters that enable end users to restrict the dashboard to displaying only certain values for these dimensions. As filter values are chosen, the component graphs, tables, and details are recalculated using only filtered values in their results. Note that if the BI Dashboard has a Range Property that is a date or is numeric, it will automatically be included as a Filter (under the Period or Range heading, respectively).



Options

Period

07/01/2001 to 10/15/2013

Filters

State (all selected)

- AZ ☑
- CA ☑
- CT ☑
- FL ☑
- GA ☑
- IL ☑
- MA ☑
- MI ☑
- MN ☑
- NC ☑
- NJ ☑
- NM ☑
- NY ☑
- OH ☑
- OK ☑

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32
MA	$352,358.58
MI	$287,995.91
MN	$364,495.00

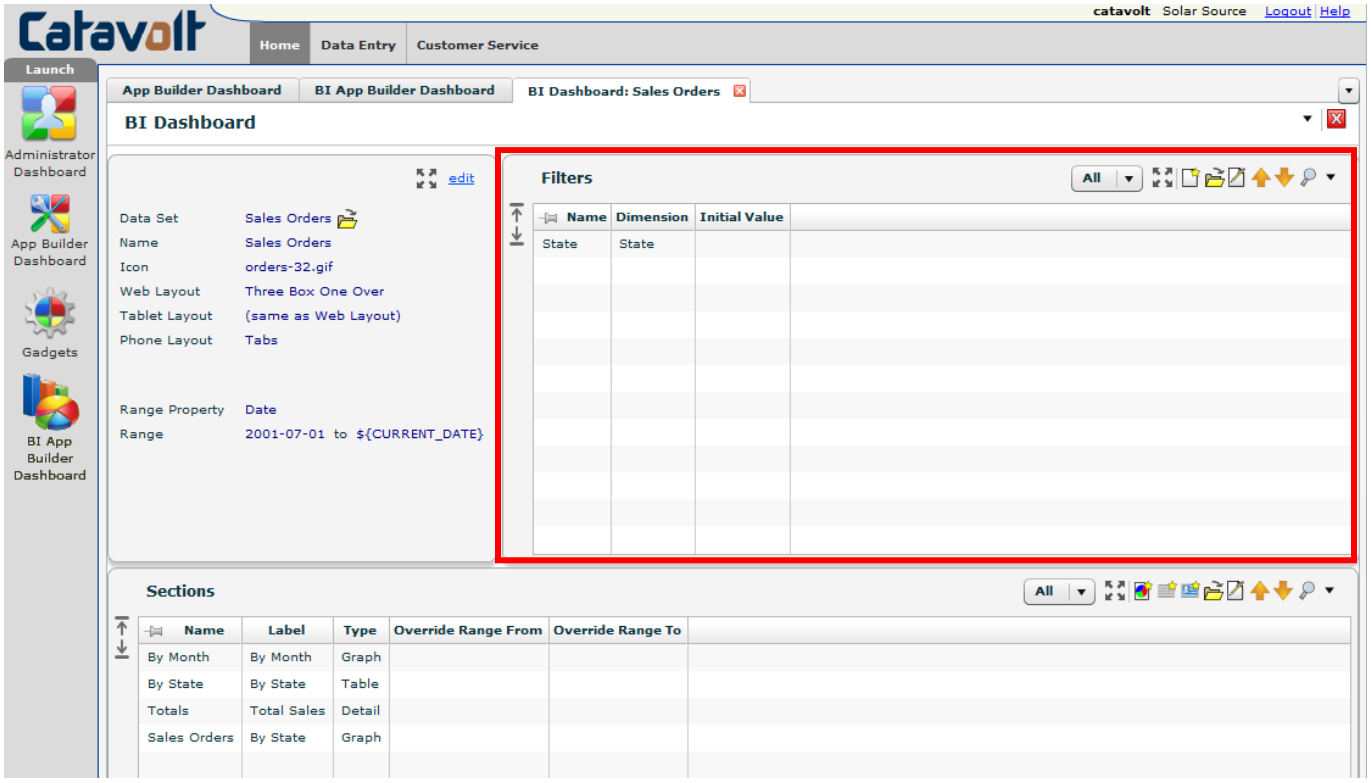
Total Sales

Amount $15,981,410.90

Figure 19: BI Dashboard with the Options sections highlighted

Accessing BI Dashboard Filters

To access the Filters defined for a particular BI Dashboard, open the BI Dashboard and go to the Filters section. Multiple Filters can be created for a BI Dashboard. They will appear in the Options panel in the same order they appear in the Filters section. To change the order, select a Filter and press the Move Up and Move Down toolbar buttons.



BI Dashboard

Data Set	Sales Orders
Name	Sales Orders
Icon	orders-32.gif
Web Layout	Three Box One Over
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
Range Property	Date
Range	2001-07-01 to ${CURRENT_DATE}

Filters

Name	Dimension	Initial Value
State	State	

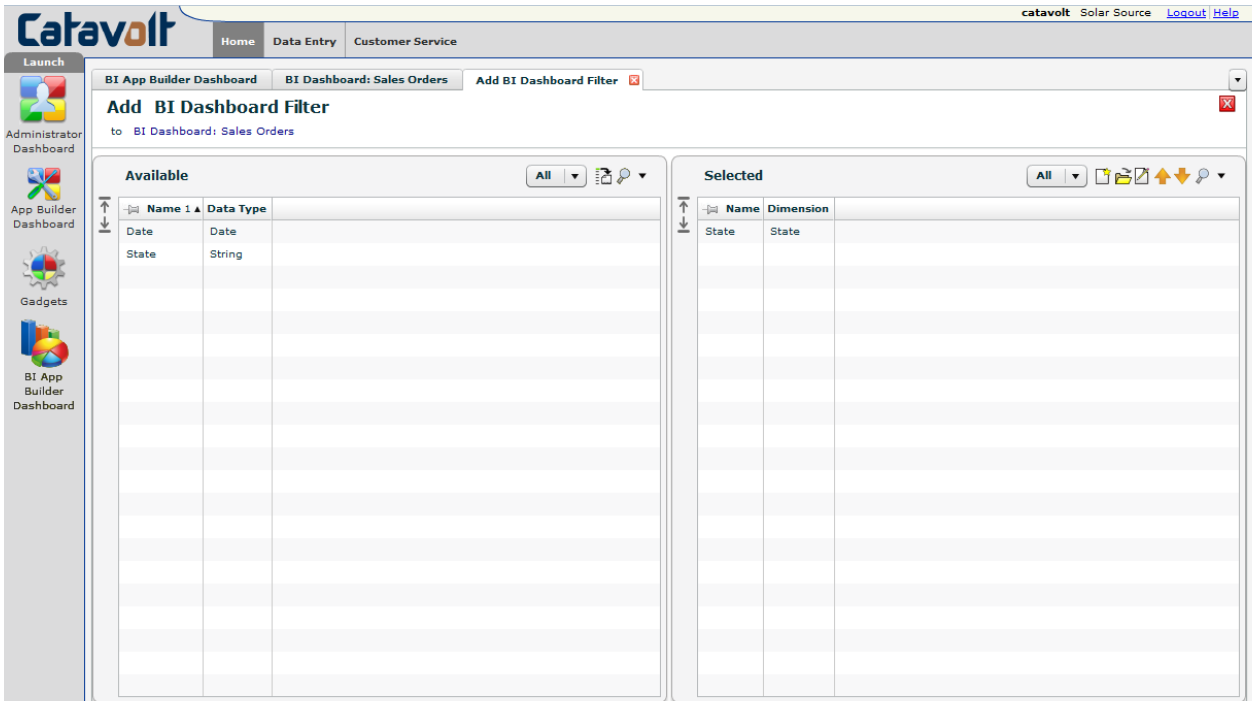
Sections

Name	Label	Type	Override Range From	Override Range To
By Month	By Month	Graph		
By State	By State	Table		
Totals	Total Sales	Detail		
Sales Orders	By State	Graph		

Figure 20: BI Dashboard details view with the Filters section highlighted

Creating BI Dashboard Filters

When adding BI Dashboard Filters, you will be presented with two lists. The Available Filters list shows all Dimensions for the Data Set. The Selected Filters list shows the dimensions that will be available to the BI Dashboard.



Available:

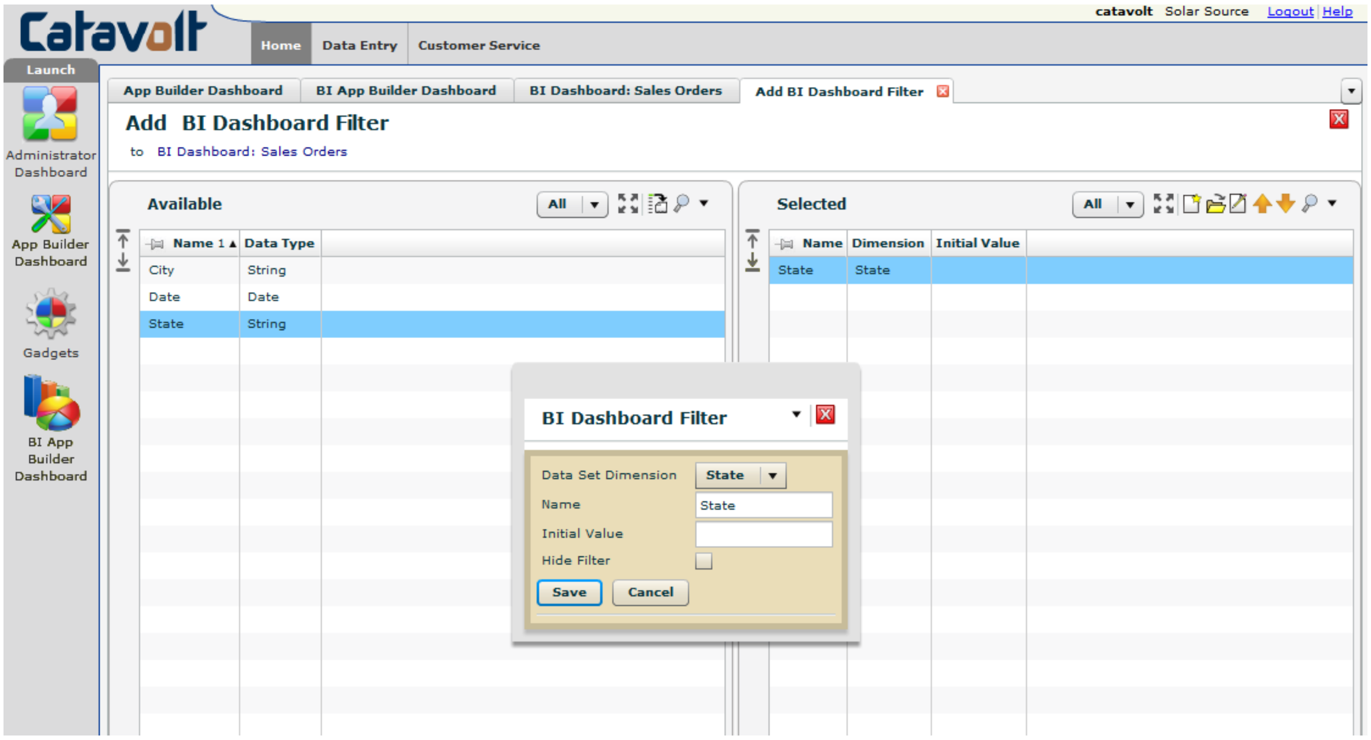
Name 1 ▲	Data Type
Date	Date
State	String

Selected:

Name	Dimension
State	State

Figure 21: BI Dashboard filter create view

You can select a single or multiple Filters and press the Add button to add them to the BI Dashboard. If you select a single filter, you will be presented with the following dialog:



Add BI Dashboard Filter

to BI Dashboard: Sales Orders

Available

Name 1 ▲	Data Type
City	String
Date	Date
State	String

Selected

Name	Dimension	Initial Value
State	State	

BI Dashboard Filter

Data Set Dimension	State
Name	State
Initial Value	
Hide Filter	□

Figure 22: BI Dashboard Filter create details view

When creating a BI Dashboard Filter, you must specify a Data Set Dimension. This specifies the dimension that this filter is over.

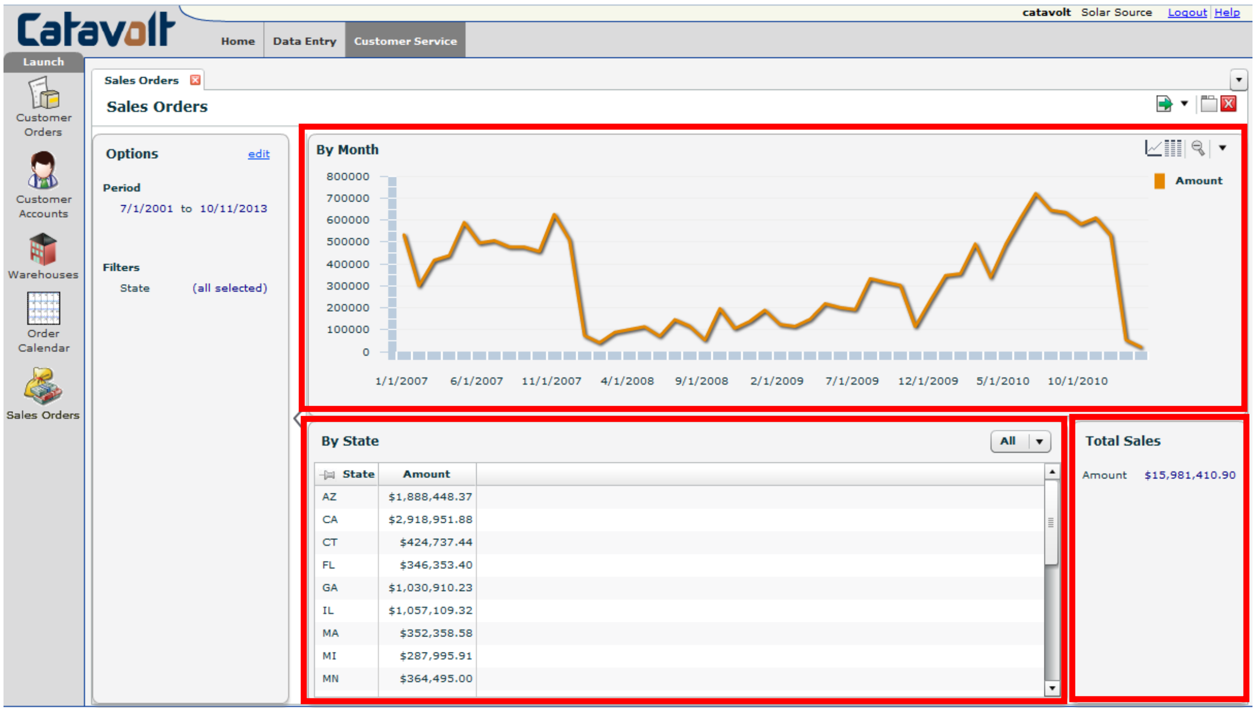
Name specifies the name of the Filter and will appear as the label on the Options pane. This defaults to the Dimension name.

Initial Value specifies the initial value of the filter when the BI Dashboard is first brought up. If the value is blank, all values from the dimension will initially be used. This option allows you to default a Filter to a single value for a dimension instead of having it initially use all values. You may use Substitution Values when specifying an Initial Value. See Appendix A: Specifying Messages and Substitution Values for more information about using Substitution Values.

Hide Filter specifies whether the user is allowed to see and change the value of the Filter when running the BI Dashboard. If Hide Filter is checked, the Filter will still be applied but will not appear on the list of Filters in the BI Dashboard. This is typically used when an Initial Value is also specified to filter the data using a value that cannot be changed by the end user.

BI Dashboard Sections

BI Dashboard Sections specify areas where data in the Data Set is rendered as a Graph, Table, or Details. Below is a BI Dashboard with 3 sections: Graph, Table, and Details. You can have multiple sections of each type within a BI Dashboard.



Sales Orders

Options edit

Period
7/1/2001 to 10/11/2013

Filters
State (all selected)

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32
MA	$352,358.58
MI	$287,995.91
MN	$364,495.00

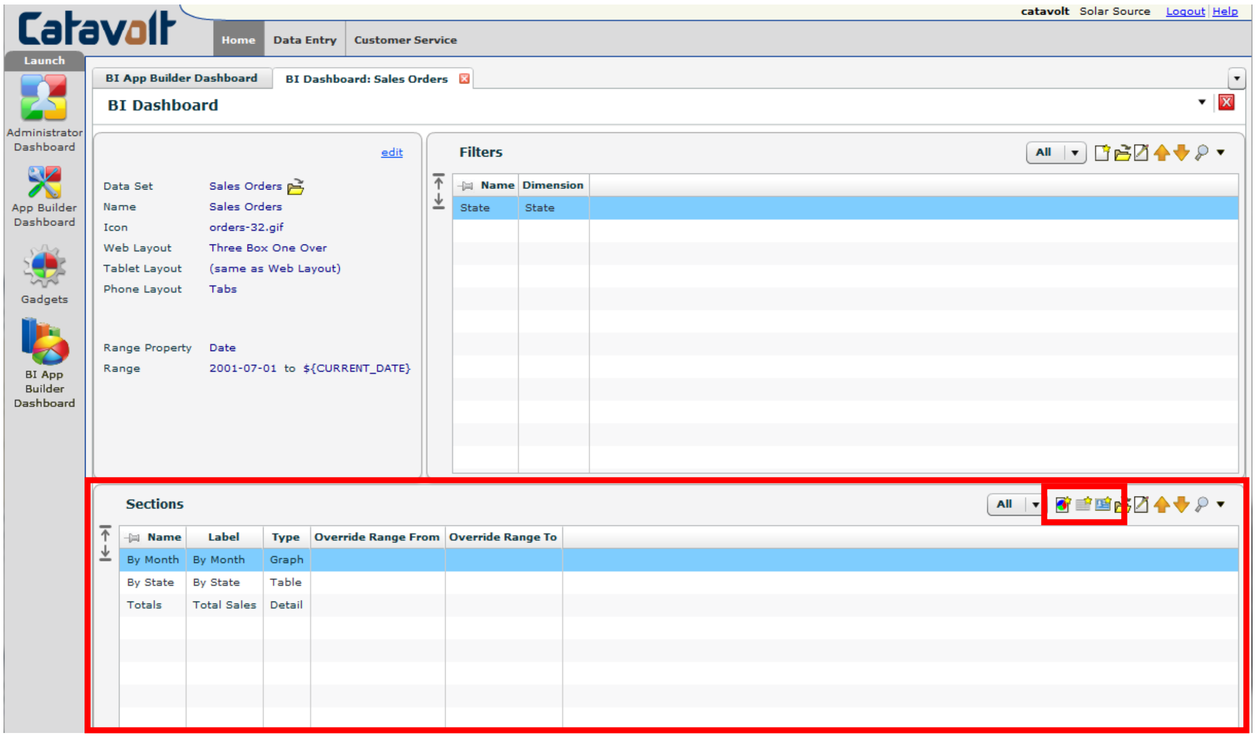
Total Sales

Amount $15,981,410.90

Figure 23: BI Dashboard with Graph, Table, and Detail sections highlighted

Accessing BI Dashboard Sections

To access the Sections defined for a particular BI Dashboard, open the BI Dashboard and go to the Sections section. Multiple Sections can be created for a BI Dashboard. They will appear in the BI Dashboard in the same order they appear in the Section section. To change the order, select a Section and press the Move Up and Move Down toolbar buttons. The sections will be laid out using the Web Layout / Tablet Layout / Phone Layout specified for the BI Dashboard.



Data Set	Sales Orders
Name	Sales Orders
Icon	orders-32.gif
Web Layout	Three Box One Over
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
Range Property	Date
Range	2001-07-01 to ${CURRENT_DATE}

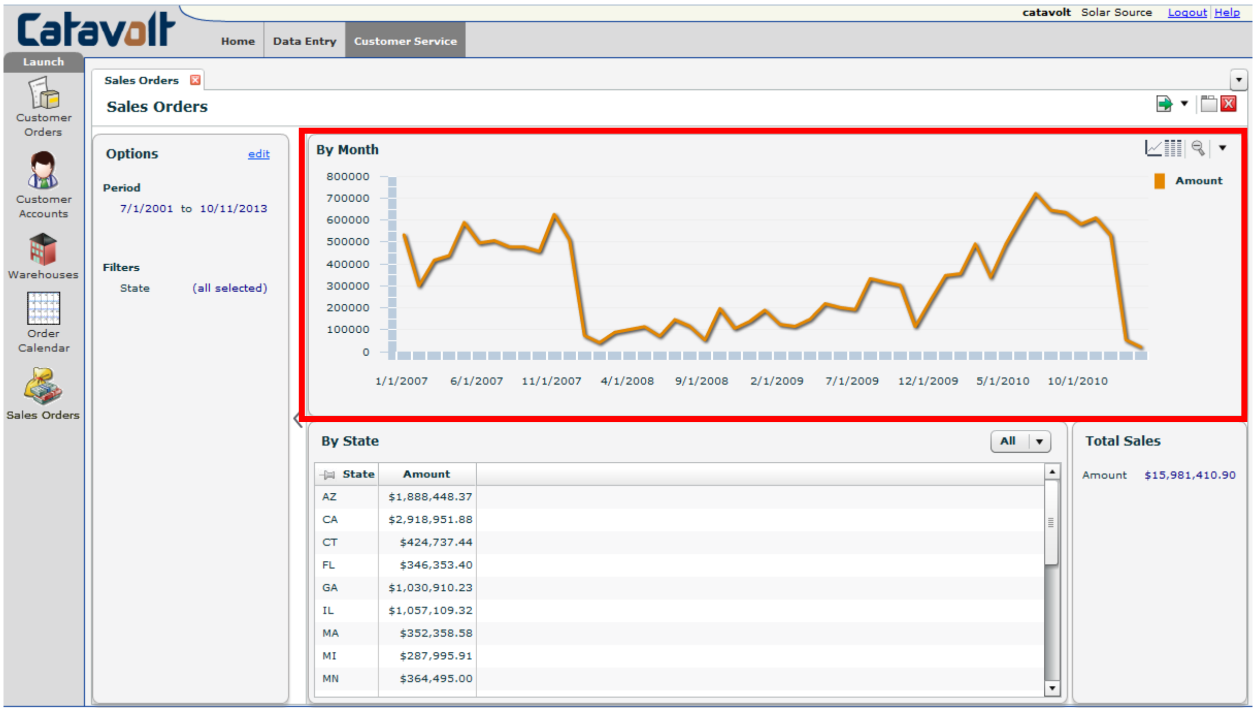
Name	Dimension
State	State

Name	Label	Type	Override Range From	Override Range To
By Month	By Month	Graph		
By State	By State	Table		
Totals	Total Sales	Detail		

Figure 24: BI Dashboard details view with the Sections section highlighted

BI Graphs

BI Graphs are sections that render data from the Data Set as a Graph.



Sales Orders

Options edit

Period
7/1/2001 to 10/11/2013

Filters
State (all selected)

By State

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32
MA	$352,358.58
MI	$287,995.91
MN	$364,495.00

Total Sales

Amount $15,981,410.90

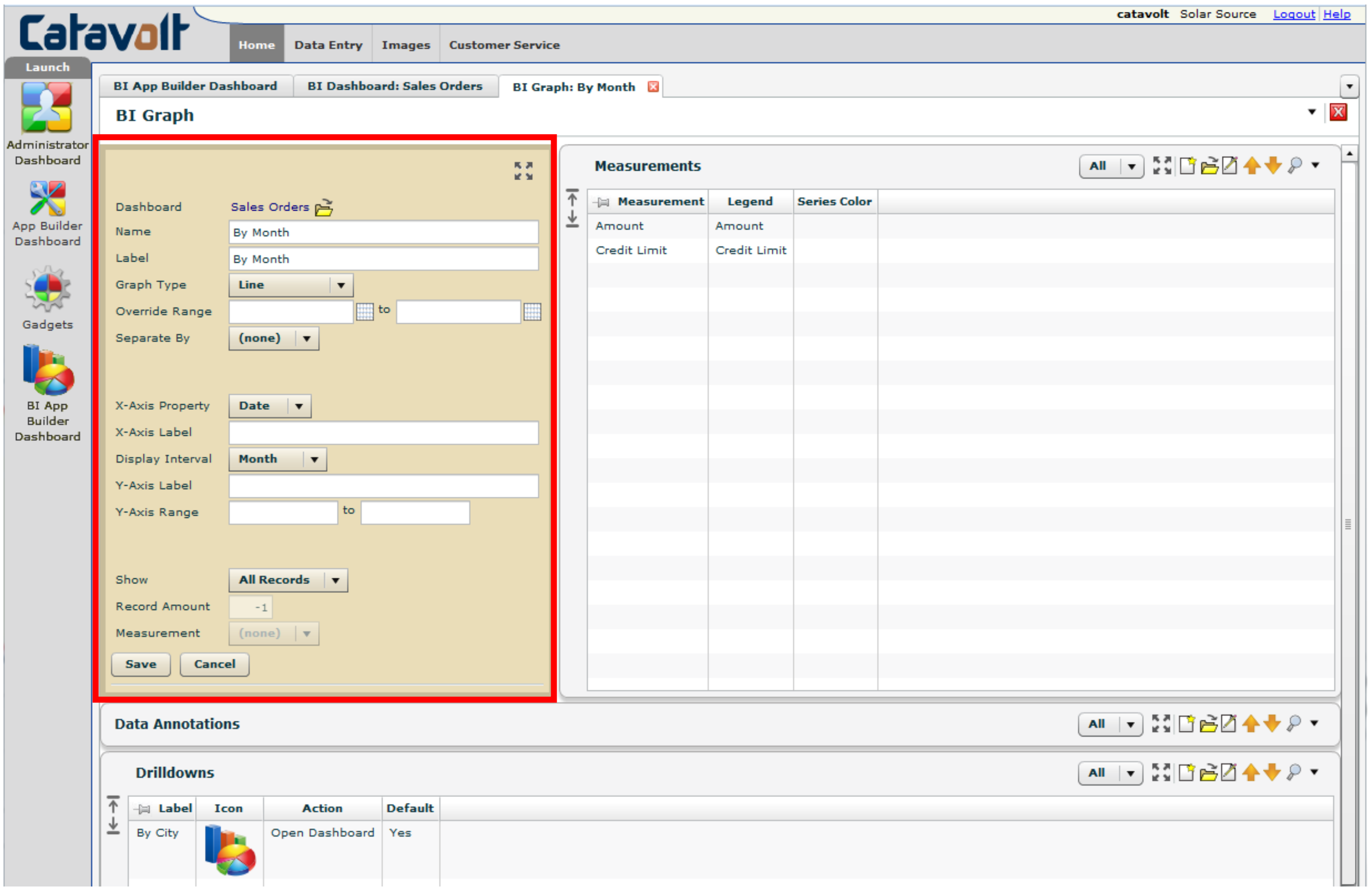
Figure 25: BI Dashboard with the Graph section highlighted

Accessing BI Graphs

To access the Graphs defined for a particular BI Dashboard, open the BI Dashboard and go to the Sections section. Multiple Graphs can be created for a BI Dashboard.

Creating BI Graphs

To create a BI Graph, choose the Add Graph menu option in the Sections section.



Dashboard	Sales Orders
Name	By Month
Label	By Month
Graph Type	Line
Override Range	to
Separate By	(none)
X-Axis Property	Date
X-Axis Label	
Display Interval	Month
Y-Axis Label	
Y-Axis Range	to
Show	All Records
Record Amount	-1
Measurement	(none)

Measurements

Measurement	Legend	Series Color
Amount	Amount	
Credit Limit	Credit Limit	

Drilldowns

Label	Icon	Action	Default
By City		Open Dashboard	Yes

Figure 26: BI Graph create details view

When creating a BI Graph, you must specify the Dashboard. This value is automatically filled in based on the BI Dashboard you are currently accessing.

Name specifies the name of the Graph. This value is used to uniquely identify the Graph to the BI Dashboard and will not be displayed to the user.

Label specifies the name that the end user will see when the Graph is displayed.

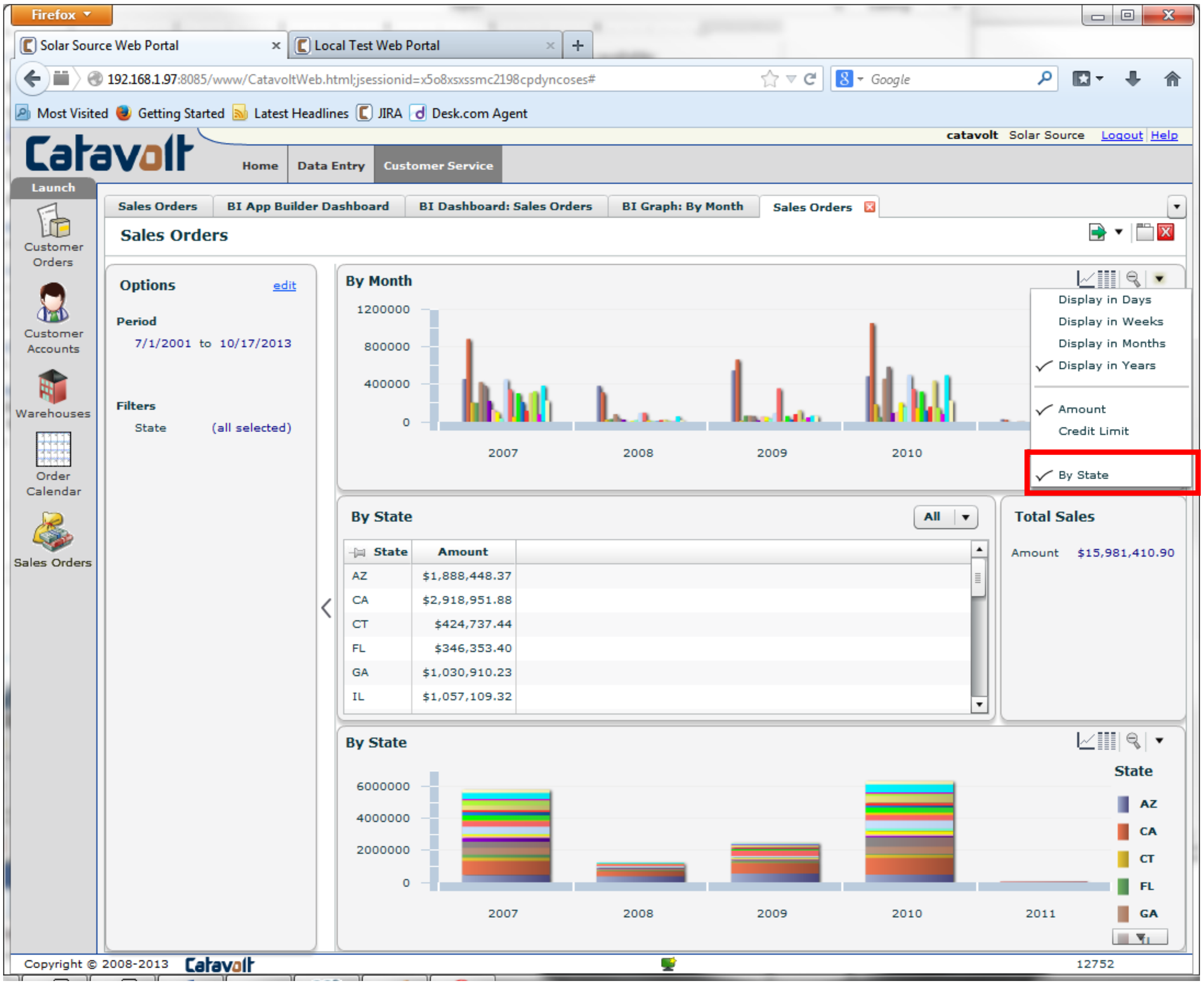
Graph Type specifies the type of graph that will be used to display data. The following options are available:

- Bar – Display as a bar graph
- Stacked Bar – Display as a stacked bar graph. The bar will be stacked based on values from the Separate By property. If you are running the BI Dashboard on a mobile device, or if the separate by property is (none), the graph will revert to displaying a Bar graph instead of a Stacked Bar.

- Line – Display as a line graph
- Pie – Display as a pie graph

Override Range allows you to use a different range for this Graph than the range being used for the BI Dashboard. Leave this value blank to use the same range as the BI Dashboard.

Separate By specifies the dimension whose values will be used to break up the graph. If Separate By is (none), a single value will be graphed (one line, one bar, etc). Specifying a Separate By property will graph multiple data points (one line per State, one bar per Region, etc). Note that the end user can change the Separate By dimension themselves at runtime via the BI Graph's dropdown menu:



Sales Orders

Options edit

Period
7/1/2001 to 10/17/2013

Filters
State (all selected)

By State

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32

Total Sales

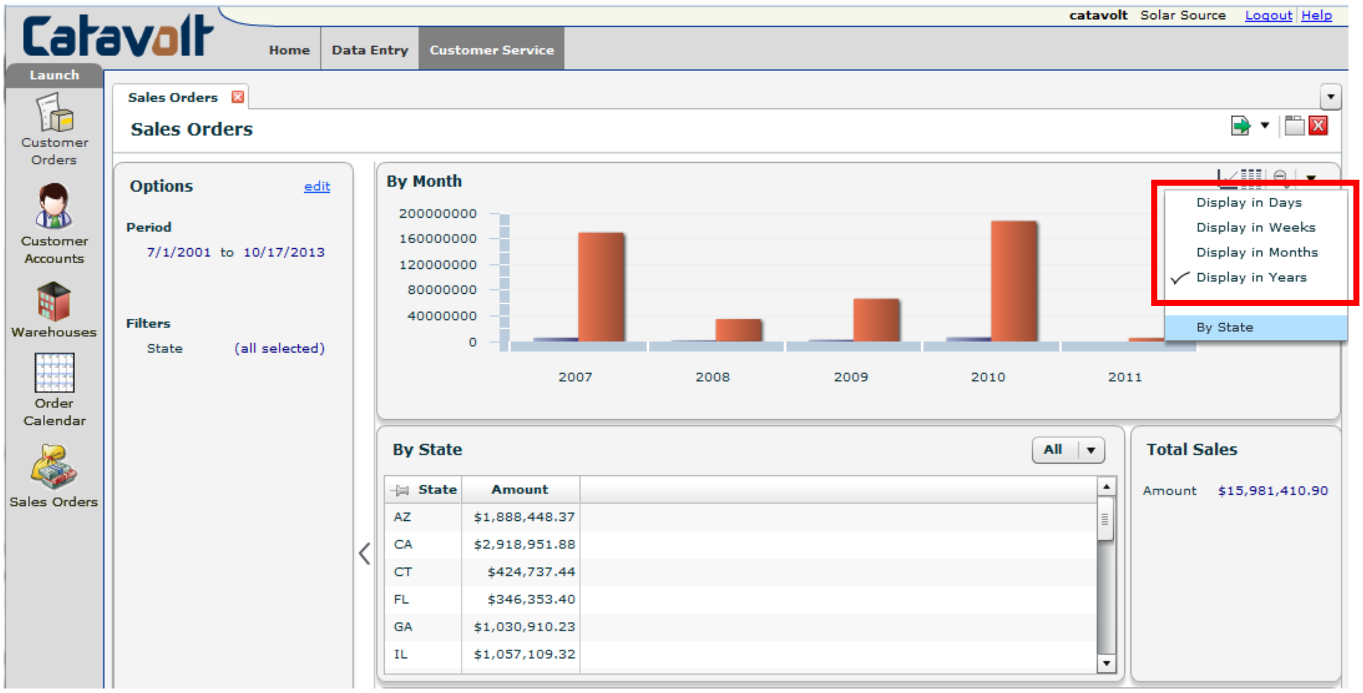
Amount $15,981,410.90

Figure 27: BI Dashboard with the Separate By options highlighted

X-Axis Property specifies the Dimension whose values will be used to map the X-Axis (horizontal line) of the Graph.

X-Axis Label specifies a label to use for the X-Axis of your graph. This option is available for all graphs except for Pie Graph.

Display Interval specifies the initial grouping time interval on the X-Axis (horizontal line) of the Graph. This value is available for X-Axis Property Dimensions that are of type Date. The available values are Day, Week, Month, Quarter, and Year. Note that the end user can change the display interval themselves at runtime via the BI Graph's dropdown menu:



Sales Orders

Options edit

Period
7/1/2001 to 10/17/2013

Filters
State (all selected)

By State

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32

Total Sales

Amount $15,981,410.90

Figure 28: BI Dashboard with the time interval options highlighted

Y-Axis Label specifies a label to use for the Y-Axis of your graph. This option is available for all graphs except for Pie Graph.

Y-Axis Range allows you to override part or all of graph's range for its Y-Axis. If these values are blank, the graph will auto-size the range to fit the graph's data. Note that you can specify one value while leaving the other blank (e.g. setting the From Range to 0 while leaving the To Range blank will anchor the bottom of the graph to 0 while letting the top of the graph float to fit the data. This option is available for all graphs except for Pie Graph.

Show specifies whether you want to calculate and display all records or a certain number of records. The available values are:

- All Records – Calculate the graph using all records
- Bottom – Calculate the graph using the bottom (lowest value) records
- Top – Calculate the graph using the top (highest value) records

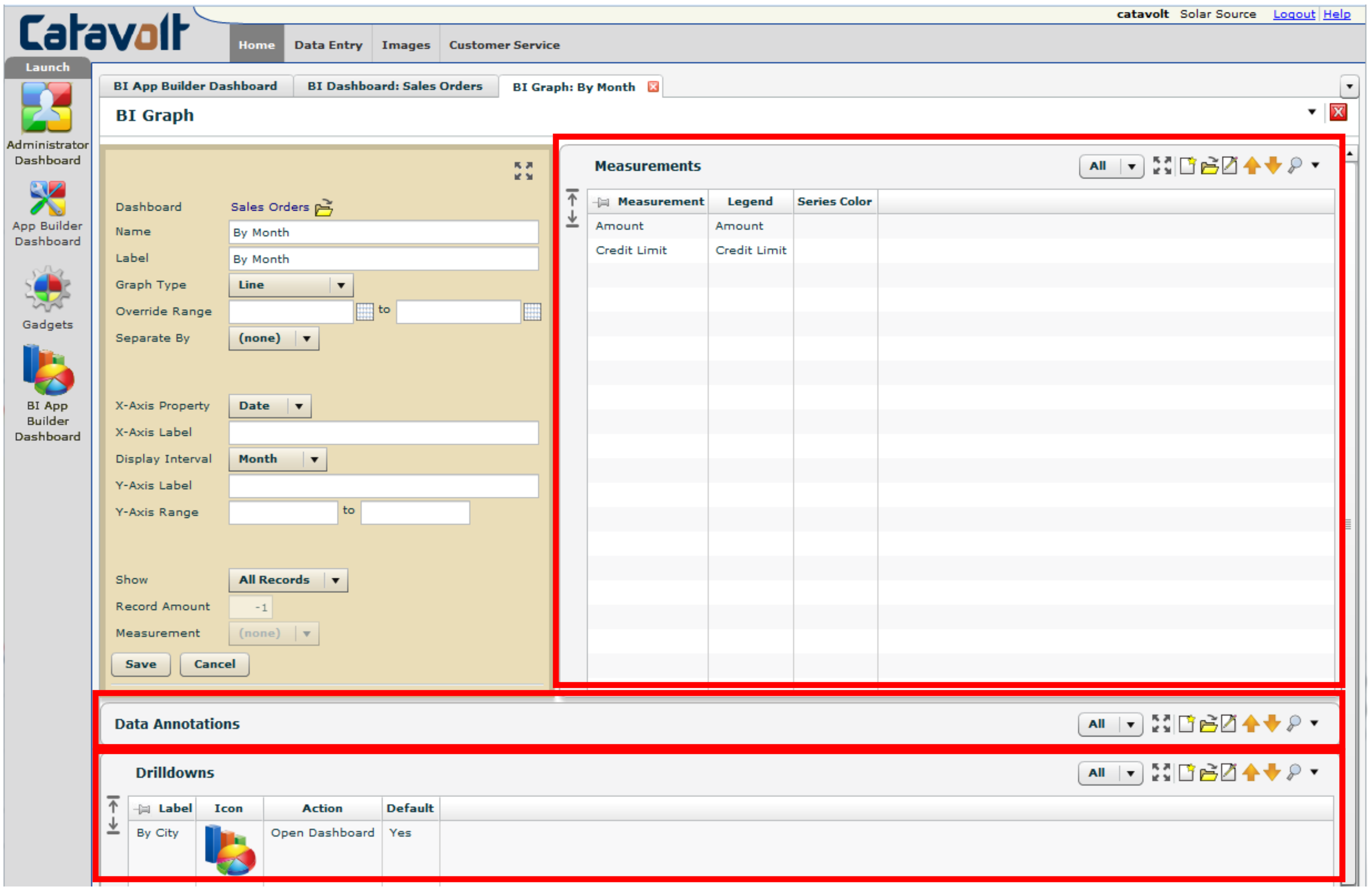
Record Amount specifies the number of records to use when calculating the Graph. This value is available when you specify Bottom or Top for the Show field.

Measurement specifies the measurement to use when calculating the Graph. This value is available when you specify Bottom or Top for the Show field. Using Show, Record Amount, and Measurement allows you to graph information such as "Show the Top 10 States by Sales Amount" or "Show the Bottom 10 Months by Total Revenue".

Series Color specifies the color you wish to use to display the value with.

BI Graph Components

BI Graphs are made up of 3 components: Measurements (a list of Measurement properties that are used to create graphs), Data Annotations (the ability to annotate data using colors, fonts and text), and Drilldowns (references to other Graphs, Tables, etc. that a user can further drill into when opening a value on the Graph).



BI Graph

Dashboard	Sales Orders
Name	By Month
Label	By Month
Graph Type	Line
Override Range	to
Separate By	(none)
X-Axis Property	Date
X-Axis Label	
Display Interval	Month
Y-Axis Label	
Y-Axis Range	to
Show	All Records
Record Amount	-1
Measurement	(none)

Measurements

Measurement	Legend	Series Color
Amount	Amount	
Credit Limit	Credit Limit	

Data Annotations

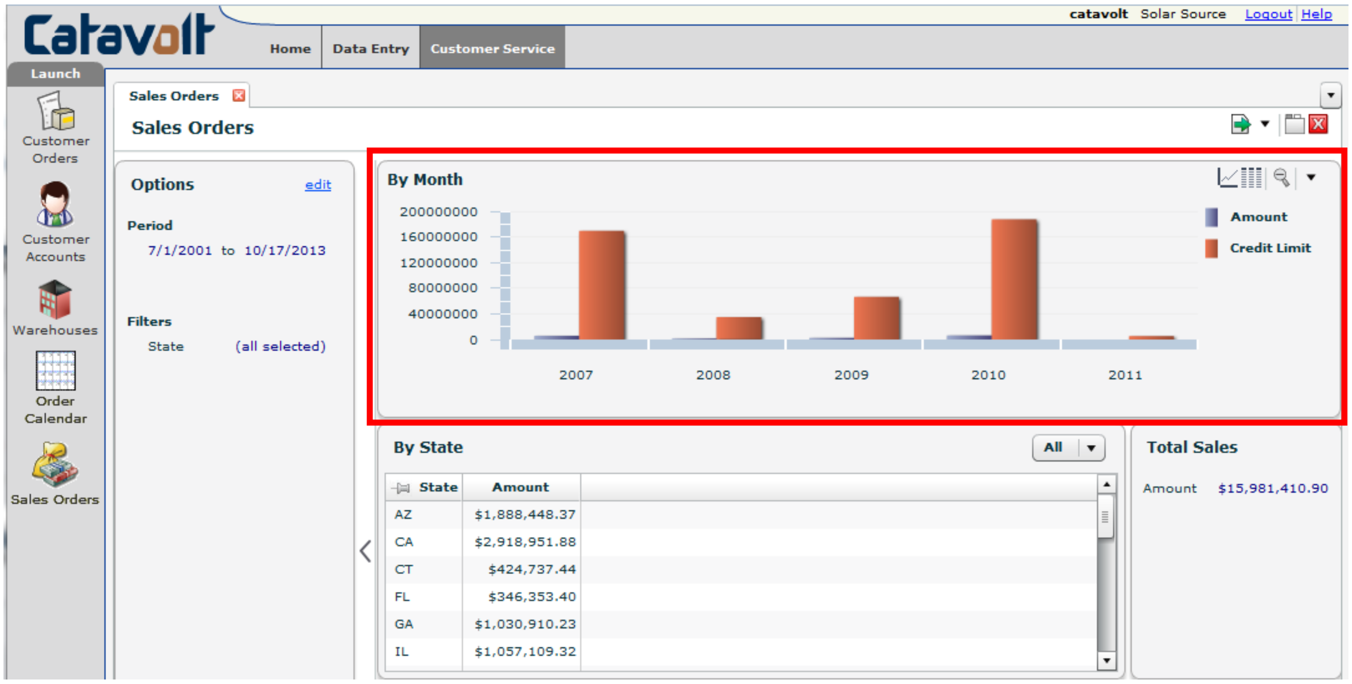
Drilldowns

Label	Icon	Action	Default
By City		Open Dashboard	Yes

Figure 29: BI Graph details view with the Measurements and Drilldowns sections highlighted

BI Graph Measurements

BI Graph Measurements represent that data values that are to be graphed. You can have multiple Measurements for a Graph. Each will be a separate value that is measured (multiple bars, multiple lines, etc.).



Options edit

Period
7/1/2001 to 10/17/2013

Filters
State (all selected)

By State

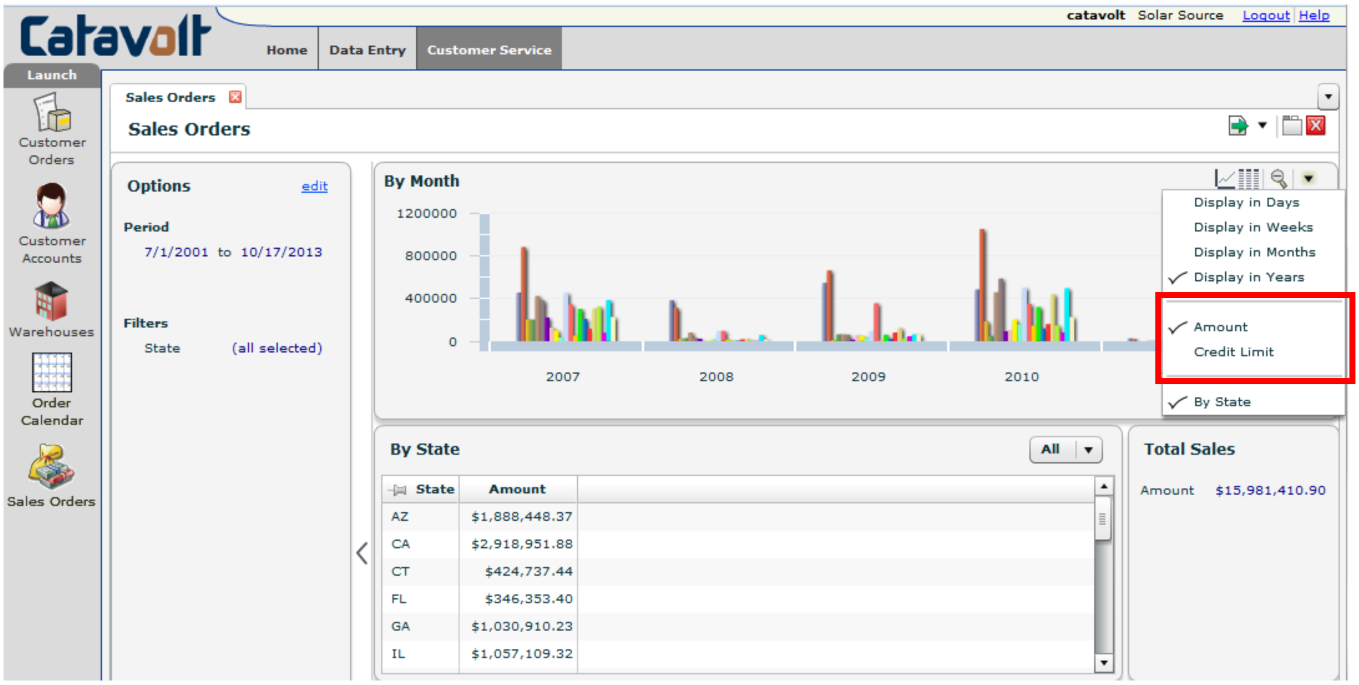
State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32

Total Sales

Amount $15,981,410.90

Figure 30: BI Dashboard with a Graph showing multiple measurements highlighted

The only exception is if you are showing a Graph with a Separate By property. In this case, since the values of the Separate By value are already showing multiple values, we only show one of the BI Graph Measurements on the graph itself. You can use the Graph's dropdown menu to choose a different measure to show.



Options edit

Period
7/1/2001 to 10/17/2013

Filters
State (all selected)

By State

State	Amount
AZ	$1,888,448.37
CA	$2,918,951.88
CT	$424,737.44
FL	$346,353.40
GA	$1,030,910.23
IL	$1,057,109.32

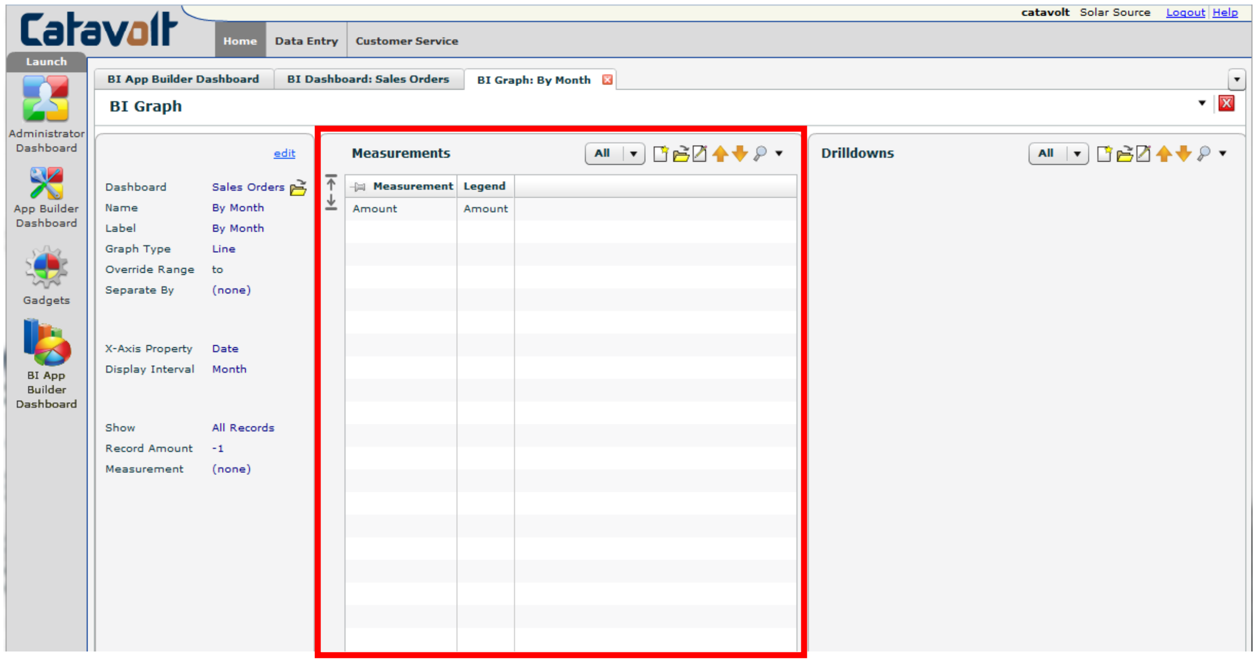
Total Sales

Amount $15,981,410.90

Figure 31: BI Dashboard with multiple measurement options highlighted

Accessing BI Graph Measurements

To access the Measurements defined for a particular BI Graph, open the BI Graph and go to the Measurements section. Multiple Measurements can be created for a BI Graph. They will appear in the Graph in the same order they appear in the Measurements section. To change the order, select a Measurement and press the Move Up and Move Down toolbar buttons.



BI Graph

Dashboard	Sales Orders
Name	By Month
Label	By Month
Graph Type	Line
Override Range	to
Separate By	(none)
X-Axis Property	Date
Display Interval	Month
Show	All Records
Record Amount	-1
Measurement	(none)

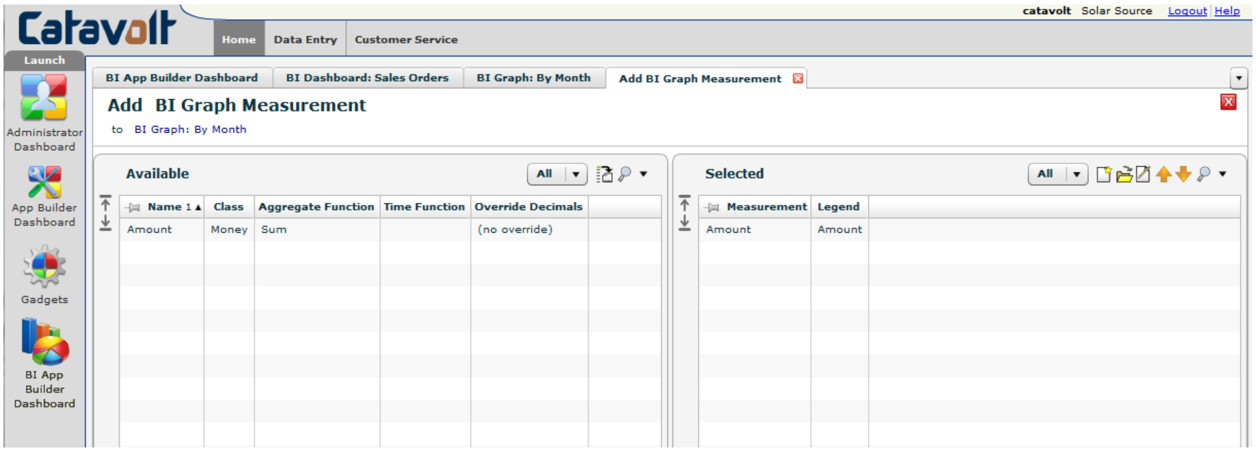
Measurements

Measurement	Legend
Amount	Amount

Figure 32: BI Graph details view with the Measurements section highlighted

Creating BI Graph Measurements

When adding BI Graph Measurements, you will be presented with two lists. The Available Measurements list shows all Measurements for the Data Set. The Selected Measurements list shows the measurements that will be used by the BI Graph.



Add BI Graph Measurement
to BI Graph: By Month

Available

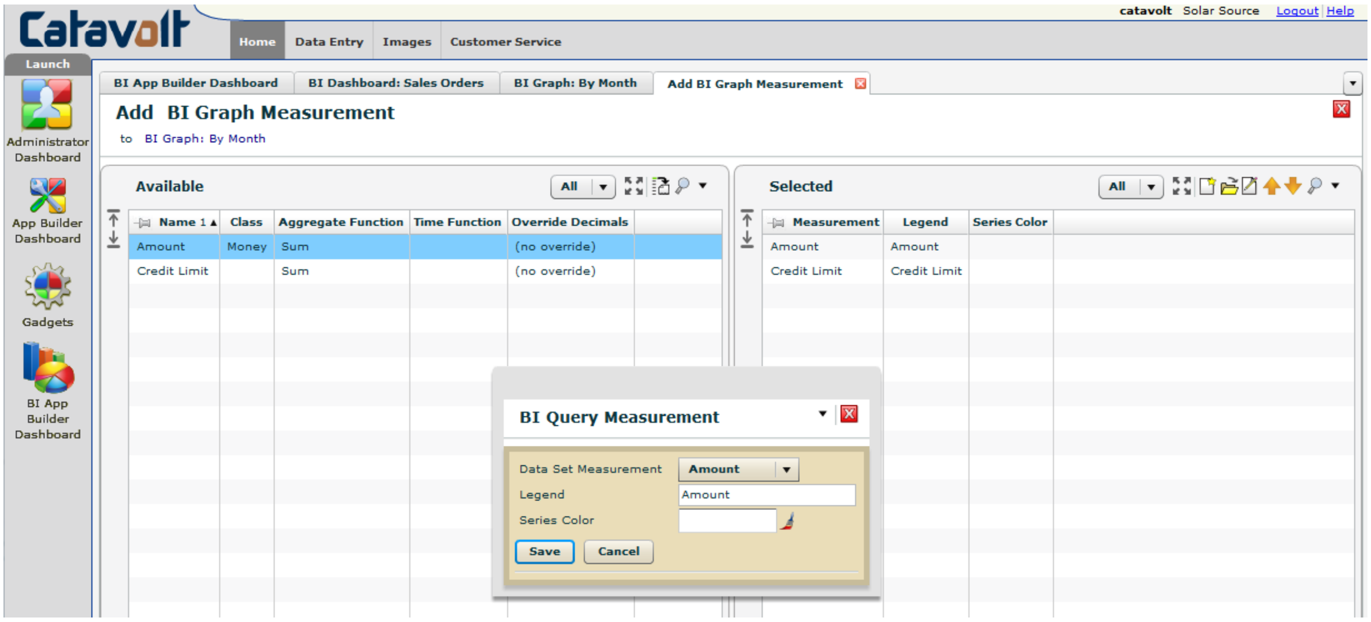
Name 1 ▲	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)

Selected

Measurement	Legend
Amount	Amount

Figure 33: Add BI Graph Measurements view

You can select a single or multiple Measurements and press the Add button to add them to the BI Graph. If you select a single measurement, you will be presented with the following dialog:



Add BI Graph Measurement
to BI Graph: By Month

Available

Name 1 ▲	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)
Credit Limit		Sum		(no override)

Selected

Measurement	Legend	Series Color
Amount	Amount	
Credit Limit	Credit Limit	

BI Query Measurement

Data Set Measurement: Amount

Legend: Amount

Series Color:

Save Cancel

Figure 34: BI Graph Measurement create details view

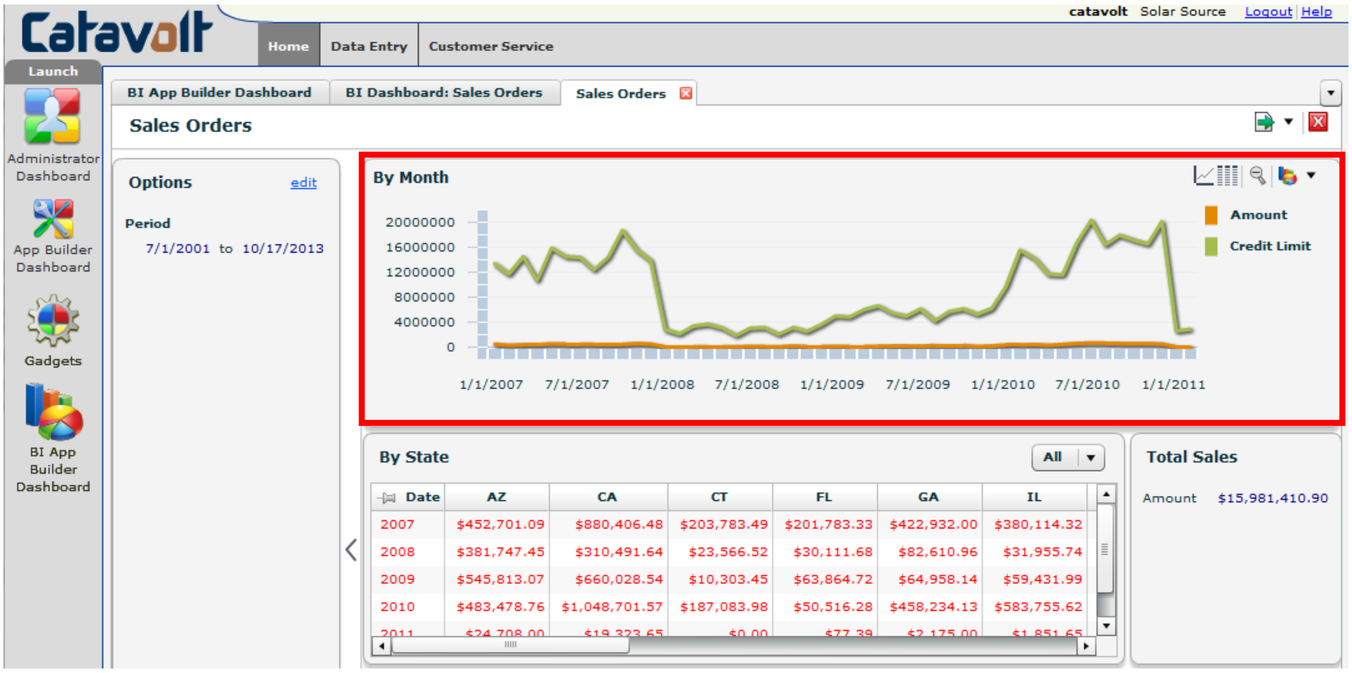
When creating a BI Graph Measurement, you must specify a Data Set Measurement. This specifies the measurement that this graph will use.

Legend specifies the legend (label) that will appear to identify the value on the Graph. This defaults to the Measurement name.

Series Color specifies the color you wish to use to display the value with.

BI Graph Data Annotations

Data Annotations allow you to change the colors used when rendering BI Graph data based on the data being displayed. The end user sees the annotations in their graph.



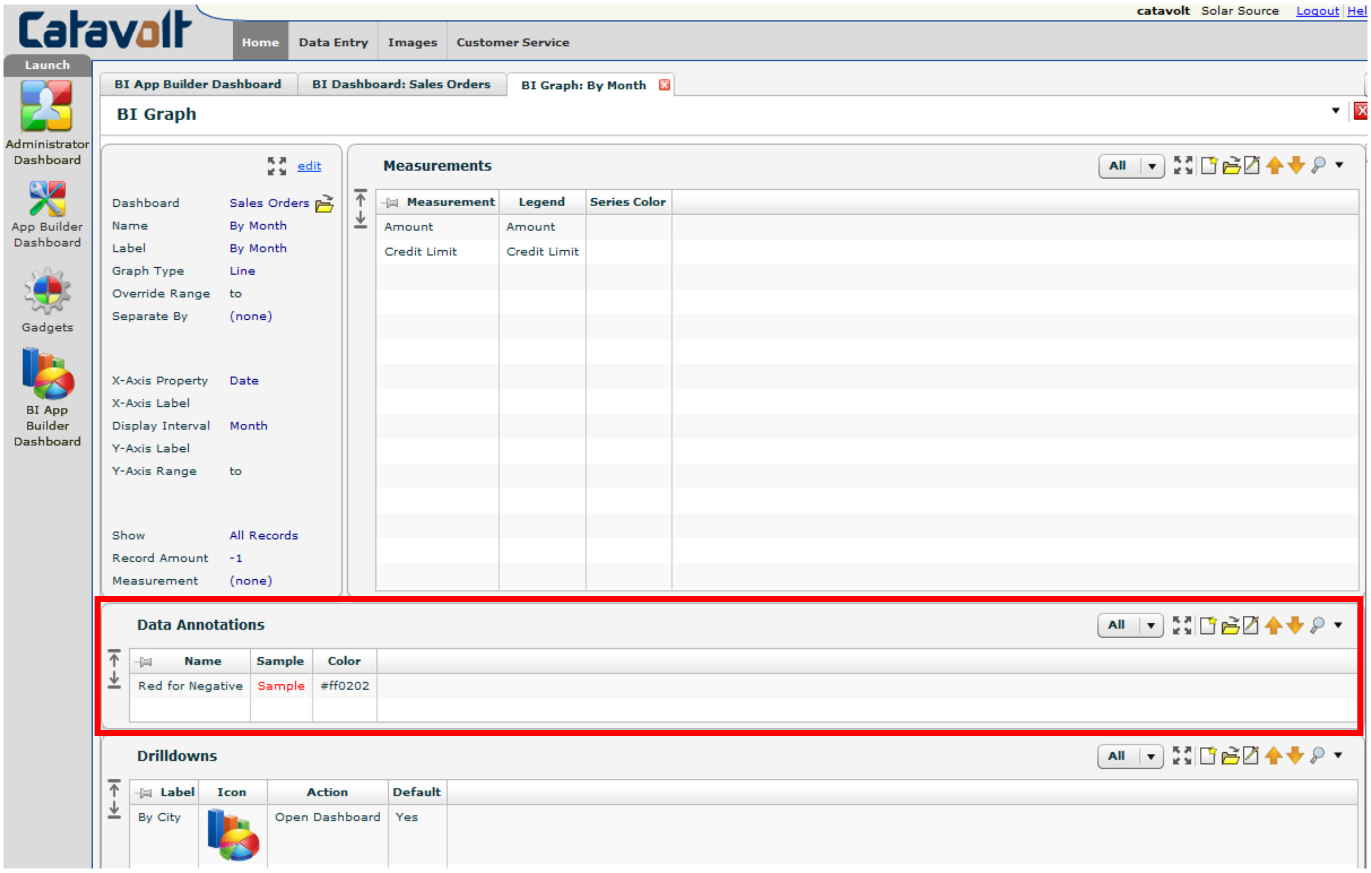
Period
7/1/2001 to 10/17/2013

Date	AZ	CA	CT	FL	GA	IL
2007	$452,701.09	$880,406.48	$203,783.49	$201,783.33	$422,932.00	$380,114.32
2008	$381,747.45	$310,491.64	$23,566.52	$30,111.68	$82,610.96	$31,955.74
2009	$545,813.07	$660,028.54	$10,303.45	$63,864.72	$64,958.14	$59,431.99
2010	$483,478.76	$1,048,701.57	$187,083.98	$50,516.28	$458,234.13	$583,755.62
2011	$24,708.00	$19,323.65	$0.00	$77.39	$2,175.00	$1,851.65

Amount $15,981,410.90

Figure 35: BI Dashboard showing a Graph with data annotations highlighted

Multiple Data Annotations are allowed for a BI Graph. Each Data Annotation can affect the graph color. When multiple Data Annotations affect the same row, the last Annotation in the list will be applied. To change the order, select a Data Annotation and press the Move Up and Move Down toolbar buttons.



BI Graph

Dashboard	Sales Orders
Name	By Month
Label	By Month
Graph Type	Line
Override Range	to
Separate By	(none)
X-Axis Property	Date
X-Axis Label	
Display Interval	Month
Y-Axis Label	
Y-Axis Range	to
Show	All Records
Record Amount	-1
Measurement	(none)

Measurements

Measurement	Legend	Series Color
Amount	Amount	
Credit Limit	Credit Limit	

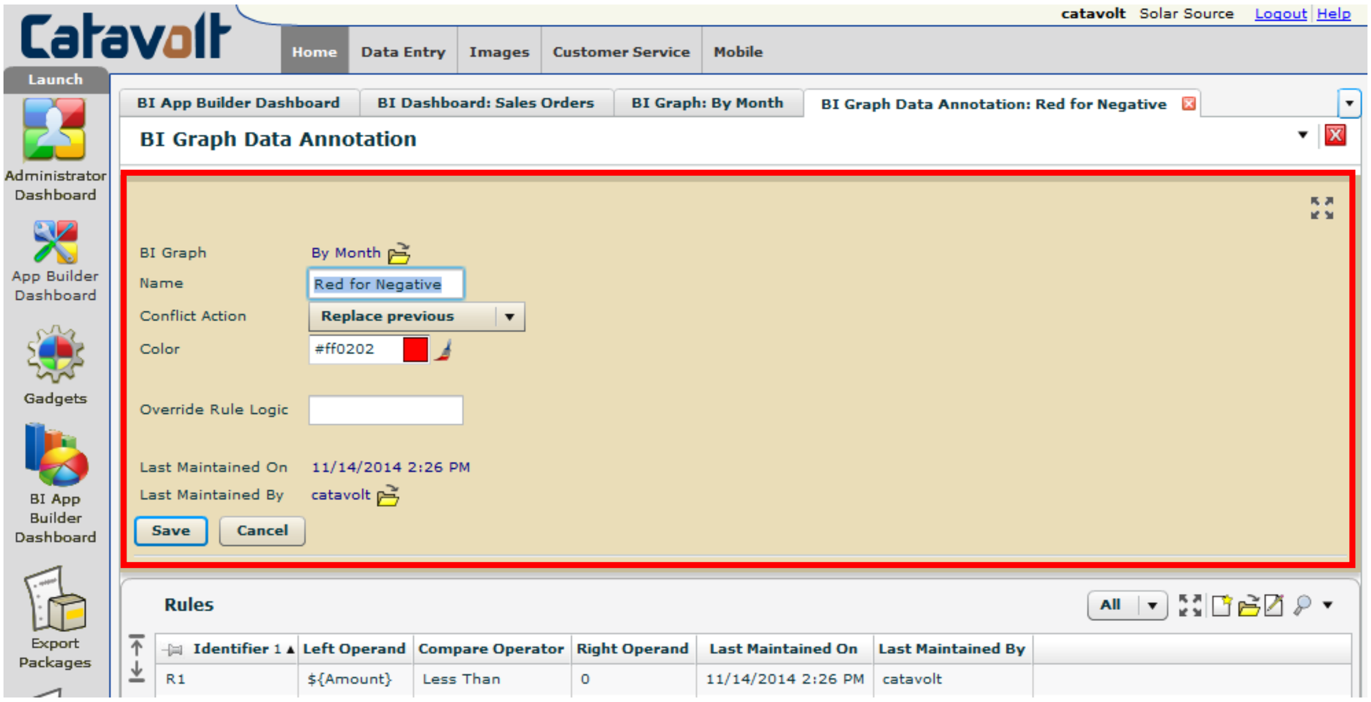
Data Annotations

Name	Sample	Color
Red for Negative	Sample	#ff0202

Drilldowns

Label	Icon	Action	Default
By City		Open Dashboard	Yes

Figure 36: BI Graph details view with Data Annotations section highlighted



BI Graph Data Annotation

BI Graph	By Month
Name	Red for Negative
Conflict Action	Replace previous
Color	#ff0202
Override Rule Logic	
Last Maintained On	11/14/2014 2:26 PM
Last Maintained By	catavolt

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand	Last Maintained On	Last Maintained By
R1	${Amount}	Less Than	0	11/14/2014 2:26 PM	catavolt

Figure 37: BI Graph Data Annotation create details view

When creating a Data Annotation, you must specify a Name. This value is used to uniquely identify the Data Annotation and is not presented to the end user.

Conflict Action specifies what should be done if multiple Data Annotations apply to a single row or a single column. As stated above, Annotations are processed in a specific sequence. Two annotations are in conflict with each other when they both pass all their rules for a row and are annotating the same column or the entire row. When annotations are in conflict, the last one in the list wins. This option allows the annotation to either be merged with previous annotations or replace previous annotations. When an annotation is merged with a previous annotation, it will assume all previous annotations' properties that it does not specify itself. For example, if one annotation sets the background color and a latter one specifies a foreground color and they are merged, the resulting annotation will have both a background and a foreground color set.

Color specifies the color you wish to change the graph line, bar, etc., if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Override Rule Logic specifies more advanced logic to evaluate the Data Annotation Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Data Annotation to be applied. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

(R1 AND R2) OR (R3 AND R4)

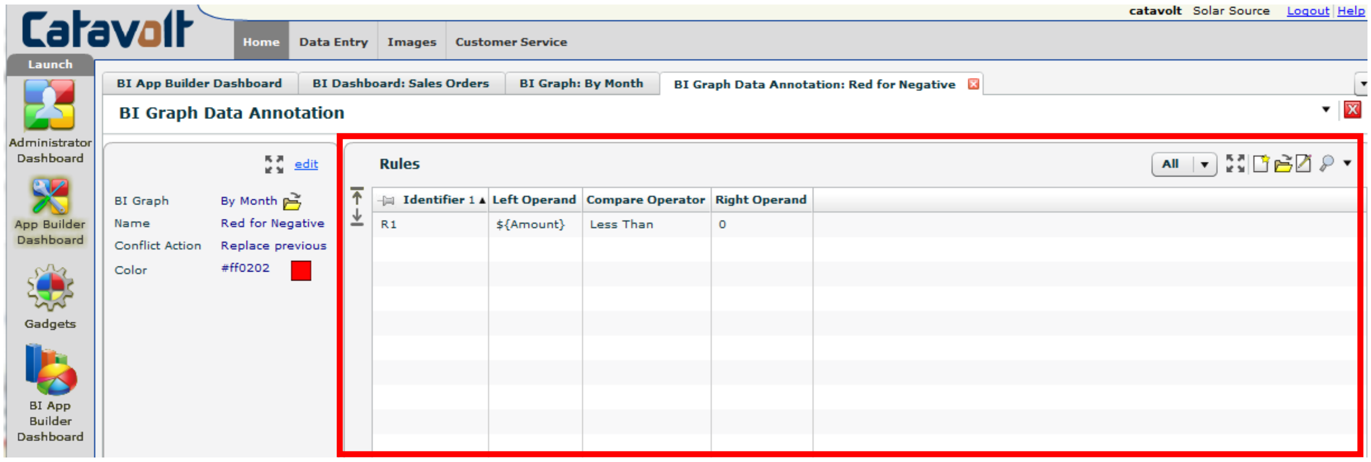
NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

BI Graph Data Annotation Rules

The Rules list on a Data Annotation can be used to control when the Data Annotation should be applied to a row of data. A Data Annotation can have zero or more rules. All rules must pass for the Data Annotation to be applied to that particular row of data. If a Data Annotation has no rules, then it is considered to pass automatically.

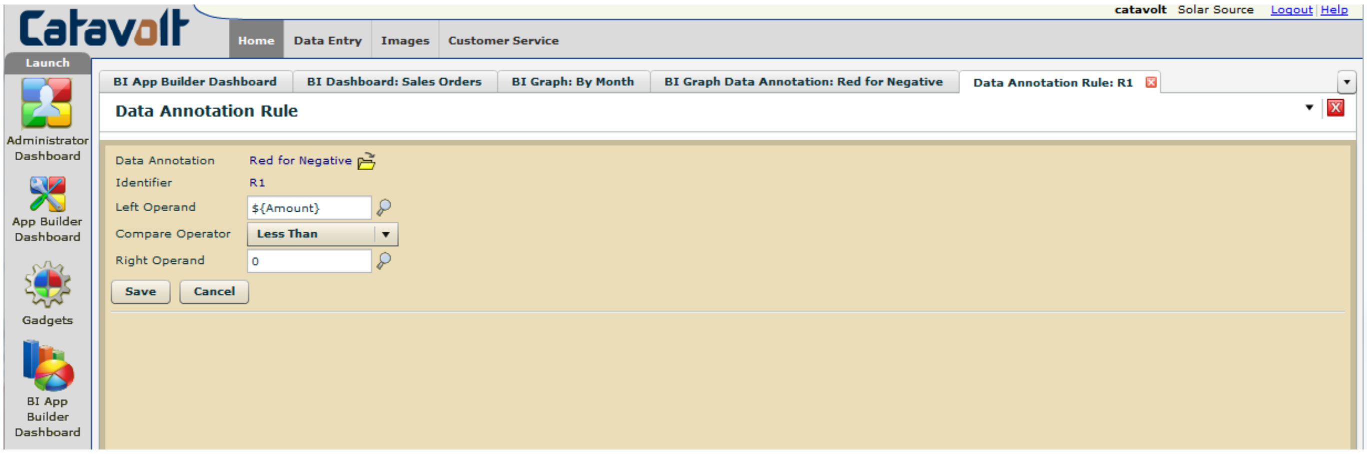


BI Graph Data Annotation

BI Graph: By Month
Name: Red for Negative
Conflict Action: Replace previous
Color: #ff0202 🟥

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand
R1	${Amount}	Less Than	0

Figure 38: BI Graph Data Annotation details view with Rules section highlighted



Data Annotation Rule

Data Annotation: Red for Negative
Identifier: R1
Left Operand: ${Amount}
Compare Operator: Less Than
Right Operand: 0

Figure 39: BI Graph Data Annotation create details view

When creating a Data Annotation Rule, Xalt will automatically create a unique Identifier for the Rule.

Left Operand and Right Operand specify the two values that will be compared with each other. You have 2 choices for operands: Constant Values (such as A or 1.00) or Substitution Values. You can enter a Substitution Value manually or press the Find button to have Xalt build a Substitution Value based on the Available Properties for the Data Object. Appendix A: Specifying Messages and Substitution Values for more information about how to create Substitution Messages for Left Operand and Right Operand.

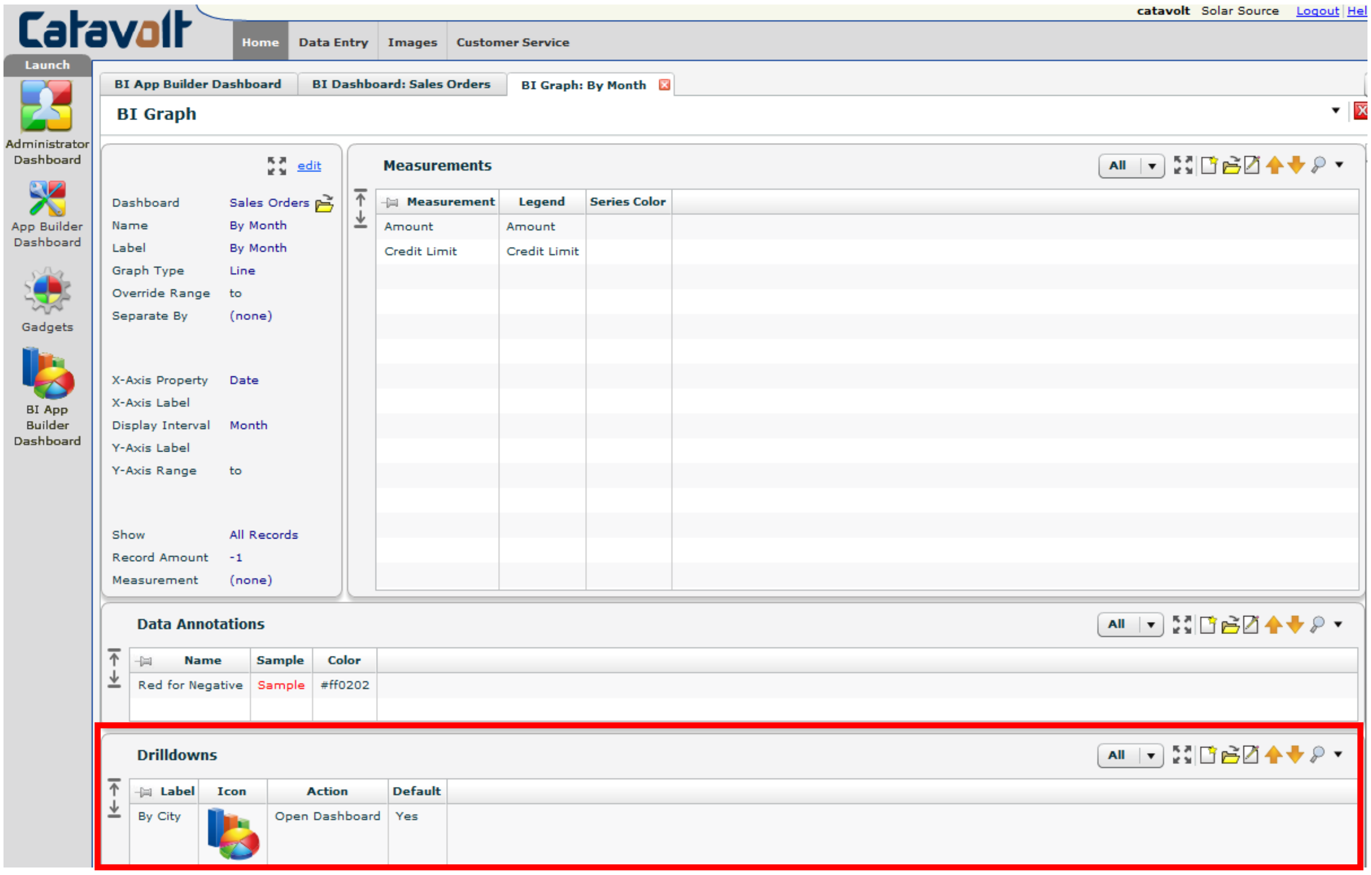
Compare Operator specifies how the two operands should be compared with each other. Options include Equal, Not Equal, Greater Than, Less Than, Greater or Equal, Less or Equal, Contains, Starts With, Ends With, Is Null, and Is Not Null.

BI Graph Drilldowns

BI Graph Drilldowns provide a way users can go from the current BI Graph and “drill-down” into another BI Dashboard, a URL, etc.

Accessing BI Graph Drilldowns

To access the Drilldowns defined for a particular BI Graph, open the BI Graph and go to the Drilldowns section. Multiple Drilldowns can be created for a BI Graph. They will appear in the Graph menu in the same order they appear in the Drilldowns section. To change the order, select a Drilldown and press the Move Up and Move Down toolbar buttons.



BI Graph

Dashboard	Sales Orders
Name	By Month
Label	By Month
Graph Type	Line
Override Range	to
Separate By	(none)
X-Axis Property	Date
X-Axis Label	
Display Interval	Month
Y-Axis Label	
Y-Axis Range	to
Show	All Records
Record Amount	-1
Measurement	(none)

Measurements

Measurement	Legend	Series Color
Amount	Amount	
Credit Limit	Credit Limit	

Data Annotations

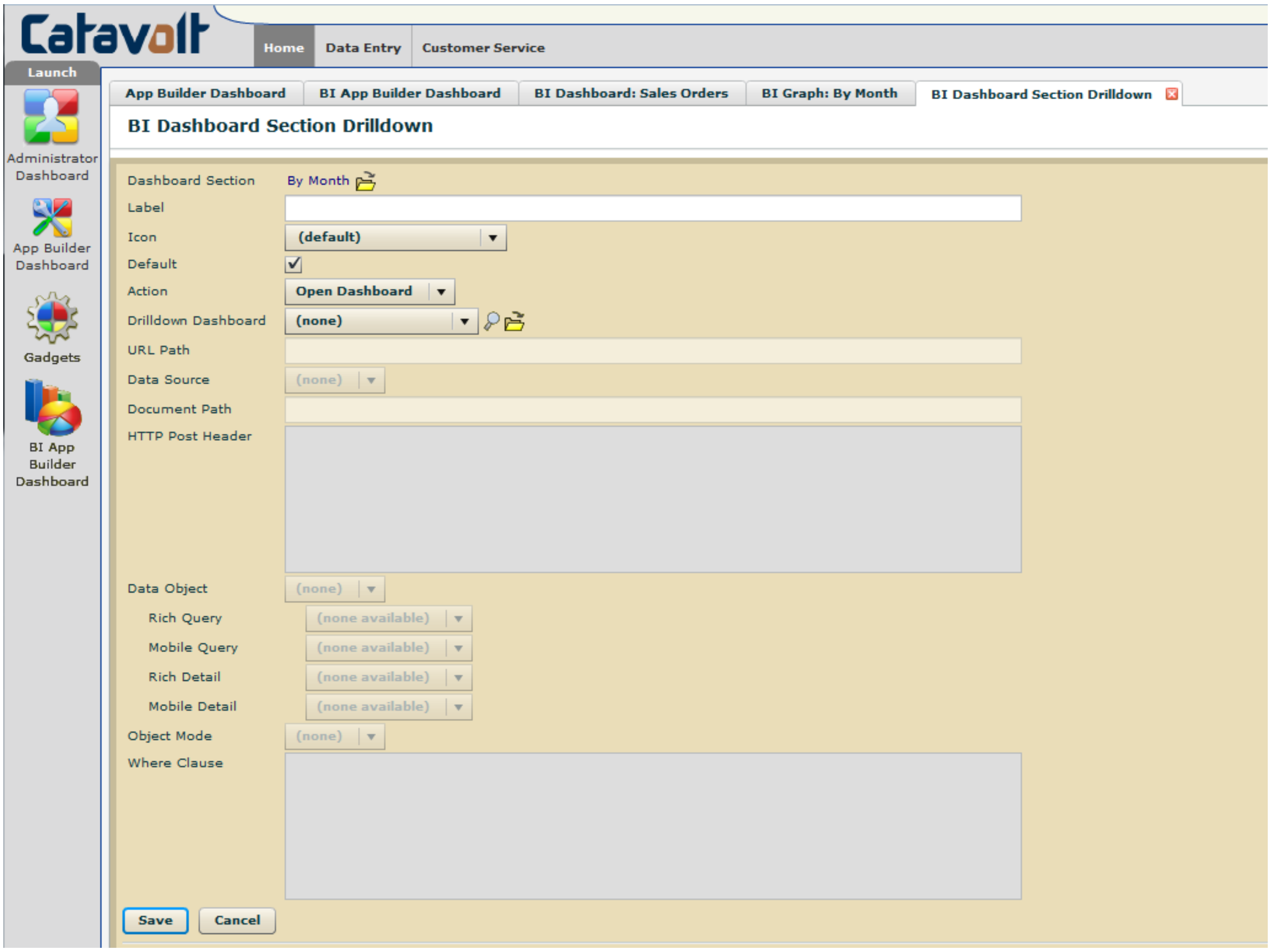
Name	Sample	Color
Red for Negative	Sample	#ff0202

Drilldowns

Label	Icon	Action	Default
By City		Open Dashboard	Yes

Figure 40: BI Graph details view with Drilldowns section highlighted

Creating BI Graph Drilldowns



BI Dashboard Section Drilldown

Dashboard Section	By Month
Label	
Icon	(default)
Default	☑
Action	Open Dashboard
Drilldown Dashboard	(none)
URL Path	
Data Source	(none)
Document Path	
HTTP Post Header	
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Object Mode	(none)
Where Clause	

Figure 41: BI Graph Drilldown create details view

When creating a BI Graph Drilldown, you must specify a Label. This specifies the name of the drilldown menu action that the graph will use.

Icon specifies the image that will be used when displaying the action on the context menu and toolbar. You can choose the default Hexagon image or any other image you have uploaded (To upload an image, choose the Upload Image menu action from the Data Sources list).

Default specifies that this is the "default" drilldown action, that is, the action that will be performed when you double-click a graph value. Drilldowns that are not the default must be selected via the graph's context menu and toolbar. You may only have one Drilldown that has been selected as default per BI Graph.

Action specifies the action that will be taken when the Drilldown's menu action is selected. The available options are:

- Open Dashboard – Open another BI Dashboard
- Open Document – Open Document will download a specified document via FTP and display it in the client (if Allow FTP is set to Yes for the Action's Data Source). You may also specify a private URL that will be accessed via the Connector gateway and then downloaded and displayed by the client

- Open URL – Open URL will open a new browser tab and display the specified URL.
- Open List – Open List will open a new tab and display a Data Object Query.
- Open Object – Open Object will open a new tab and display a Data Object Detail in either read or update mode.

Drilldown Dashboard specifies the BI Dashboard that will be displayed if your Action is Open Dashboard. Any BI Dashboard Filters in common between the BI Dashboards will be passed down to the Drilldown Dashboard from the current BI Dashboard.

URL Path specifies the public URL that will be displayed if your Action is Open URL.

Data Source specifies the Data Source that will be used to FTP the document to display if your Action is Open Document. Note that FTP must be enabled on the Data Source you select.

Document Path allows you to specify the path and name of the document that is to be downloaded if your Action is Open Document. The value of this field will be appended to the FTP Path field in the Data Source in order to get the full path to be downloaded via FTP from the back end system. If a URL is specified for Open Document, the target of the URL (an image, PDF, etc.) will be downloaded from the Connector gateway and sent to the client.

HTTP Post Header allows you to open a URL via an HTTP Post instead of an HTTP Get. You can specify parameters in the HTTP Post Header to use when making the HTTP Post call.

Data Object specifies which Data Object you want to display either a list or object of or perform an action against. This value is only allowed if the Action is Open Object or Open List.

Rich Query specifies which Query you want to display on the list when using a rich client. This value is only allowed if the Action is Open List.

Mobile Query specifies which Query you want to display on the list when using a mobile client. This value is only allowed if the Action is Open List.

Rich Detail specifies which Detail you want to display on the object when using a rich client. This value is only allowed if the Action is Open Object.

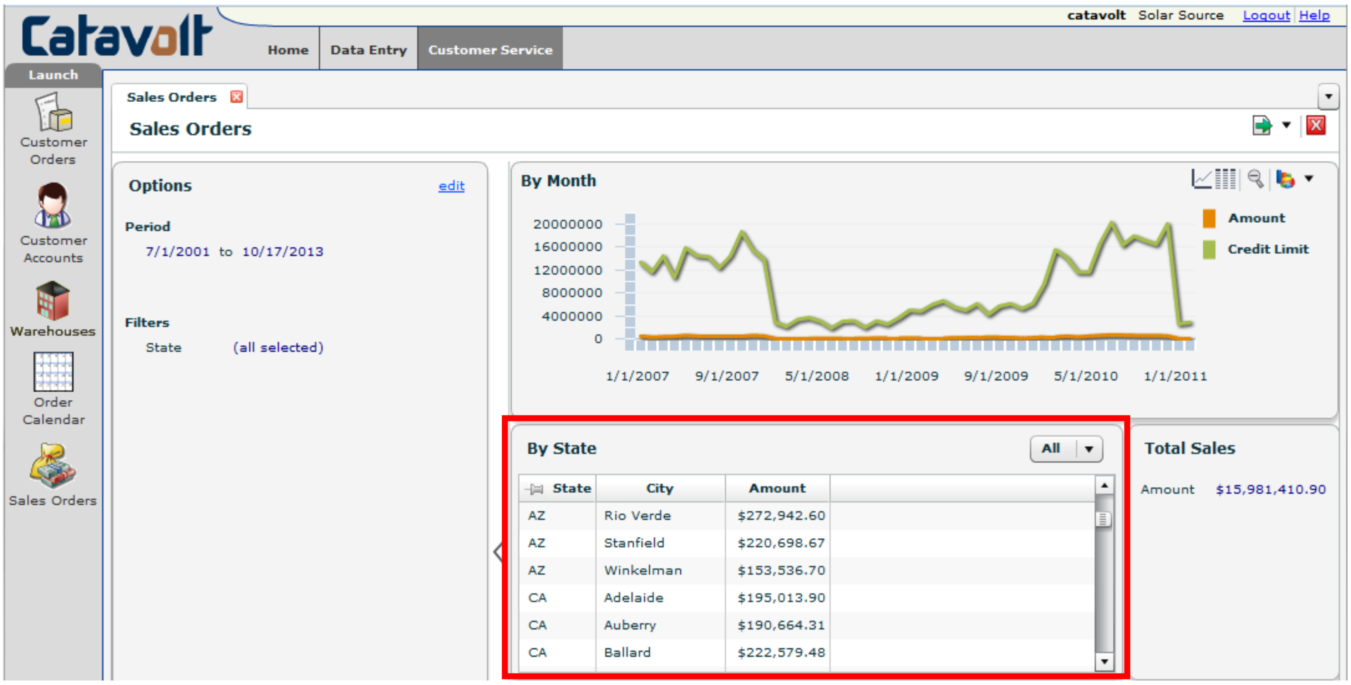
Mobile Detail specifies which Detail you want to display on the object when using a mobile client. This value is only allowed if the Action is Open Object.

Object Mode specifies to display the Detail in read-only or update mode when displaying an object. This value is only allowed if the Action is Open Object.

Where Clause specifies extra criteria to use when querying an object. If the Action is Open Object, then Where Clause must be written to return a single record when it is performed (same as when creating a Dashboard Launcher for a Data Object). If the Action is Open List, Where Clause can be optionally specified to further restrict the list of records returned. This value is only allowed if the Action is Open Object or Open List. You can access the Graph's X-Axis Property in the Where Clause. For example, if the X-Axis Property is "Date", you can use ${Date} in the Where Clause. Note that if you have a Display Interval other than Day, you may need to manipulate the values passed to you. For example, if you have a Display Interval of Month and a Postgresql database, your Where Clause would be the following in order to see all records for the specified month: EXTRACT(YEAR FROM order_date) = EXTRACT(YEAR FROM DATE '${Date}') AND EXTRACT(MONTH FROM order_date) = EXTRACT(MONTH FROM DATE '${Date}')

BI Tables

BI Tables allow you to present data from your Data Set in tabular format. A BI Table consists of one or more dimensions combined with one or more measurements. Each combination of values from the dimensions will result in a row in the BI Table. For example, in the table below, the dimensions included are State and City. Each combination of State and City is a row in the table. The Measurements in the table are calculated separately for each State/City combination.



Period
7/1/2001 to 10/17/2013

Filters
State (all selected)

State	City	Amount
AZ	Rio Verde	$272,942.60
AZ	Stanfield	$220,698.67
AZ	Winkelman	$153,536.70
CA	Adelaide	$195,013.90
CA	Auberry	$190,664.31
CA	Ballard	$222,579.48

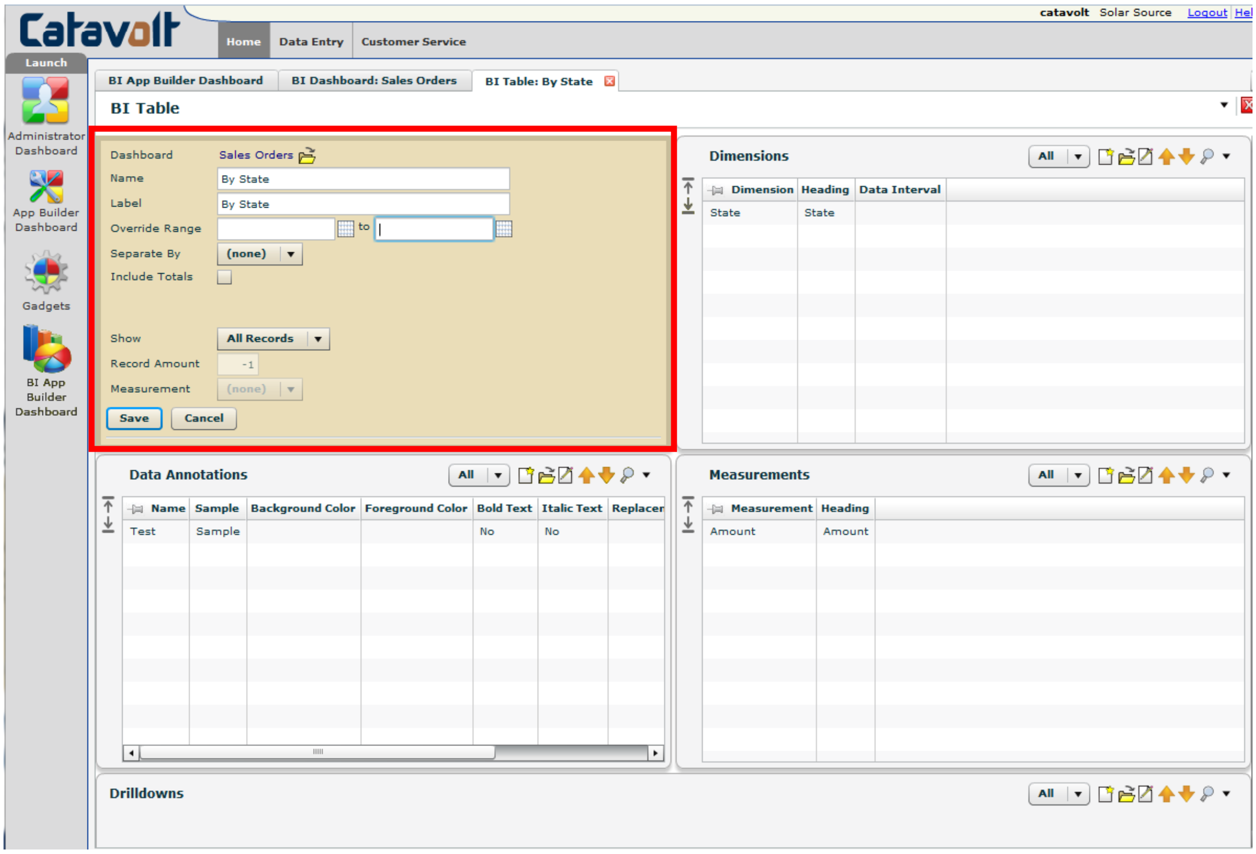
Amount $15,981,410.90

Figure 42: BI Dashboard with a Table section highlighted

Accessing BI Tables

To access the Tables defined for a particular BI Dashboard, open the BI Dashboard and go to the Sections section. Multiple Tables can be created for a BI Dashboard.

Creating BI Tables



Dashboard	Sales Orders
Name	By State
Label	By State
Override Range	to
Separate By	(none)
Include Totals	□
Show	All Records
Record Amount	-1
Measurement	(none)

Dimensions

Dimension	Heading	Data Interval
State	State	

Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacer
Test	Sample			No	No	

Measurements

Measurement	Heading
Amount	Amount

Figure 43: BI Table create details view

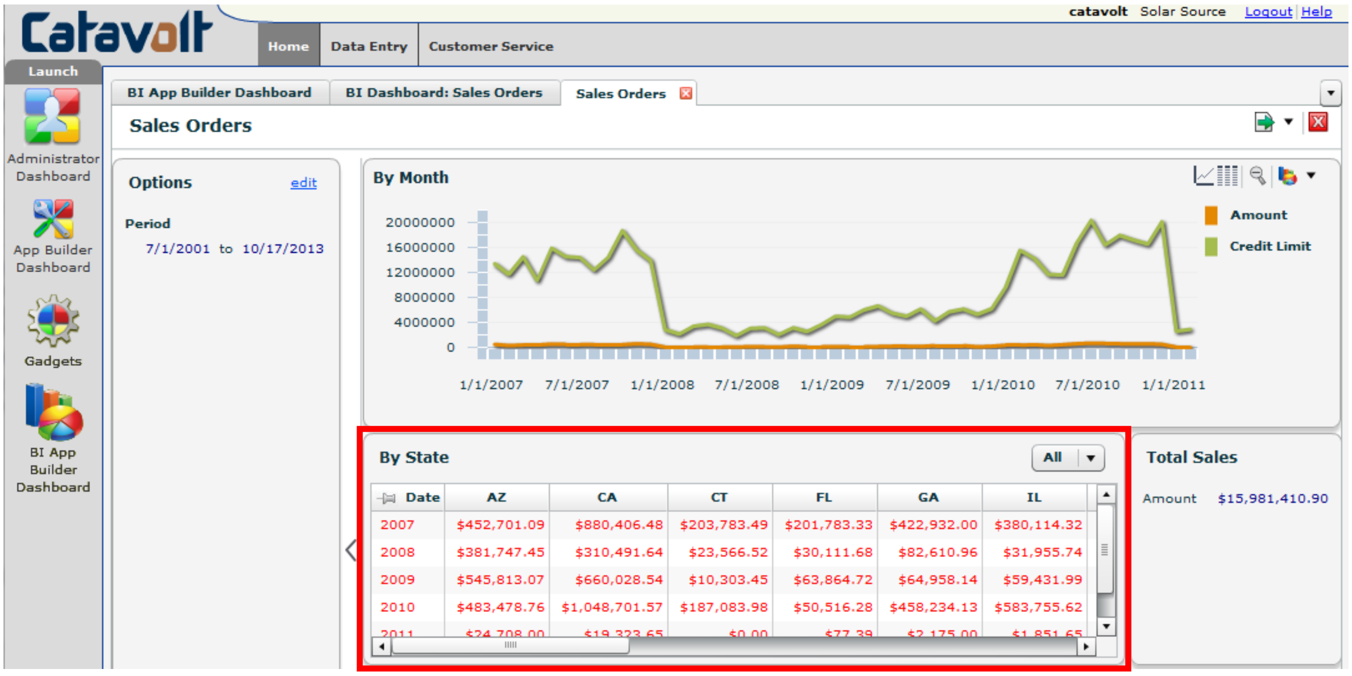
When creating a BI Table, you must specify the Dashboard. This value is automatically filled in based on the BI Dashboard you are currently accessing.

Name specifies the name of the Table. This value is used to uniquely identify the Table to the BI Dashboard and will not be displayed to the user.

Label specifies the name that the end user will see when the Table is displayed.

Override Range allows you to use a different range for this Table than the range being used for the BI Dashboard. Leave this value blank to use the same range as the BI Dashboard.

Separate By specifies the dimension whose values will be used to break up the graph. If Separate By is (none), a single column per measurement will be graphed. Specifying a Separate By property will show multiple columns, one for each dimension value. Below is an example of a BI Table where the Dimension is Date (by Year), and the Separate By is State. Note that each row represents a Year, and each column is a State. The cell value is the Measurement (sales amount) for each State per Year:



Sales Orders

Period
7/1/2001 to 10/17/2013

By State

Date	AZ	CA	CT	FL	GA	IL
2007	$452,701.09	$880,406.48	$203,783.49	$201,783.33	$422,932.00	$380,114.32
2008	$381,747.45	$310,491.64	$23,566.52	$30,111.68	$82,610.96	$31,955.74
2009	$545,813.07	$660,028.54	$10,303.45	$63,864.72	$64,958.14	$59,431.99
2010	$483,478.76	$1,048,701.57	$187,083.98	$50,516.28	$458,234.13	$583,755.62
2011	$24,708.00	$19,323.65	$0.00	$77.39	$2,175.00	$1,851.65

Total Sales

Amount $15,981,410.90

Figure 44: BI Dashboard with Table using Separate By property highlighted

Include Totals specifies whether or not to include totals with the table. If checked, a Totals line will be added at the bottom to include column totals. If you have specified a Separate By dimension, a Totals column on the right will also be added to include row totals.

Show specifies whether you want to calculate and display all records or a certain number of records. The available values are:

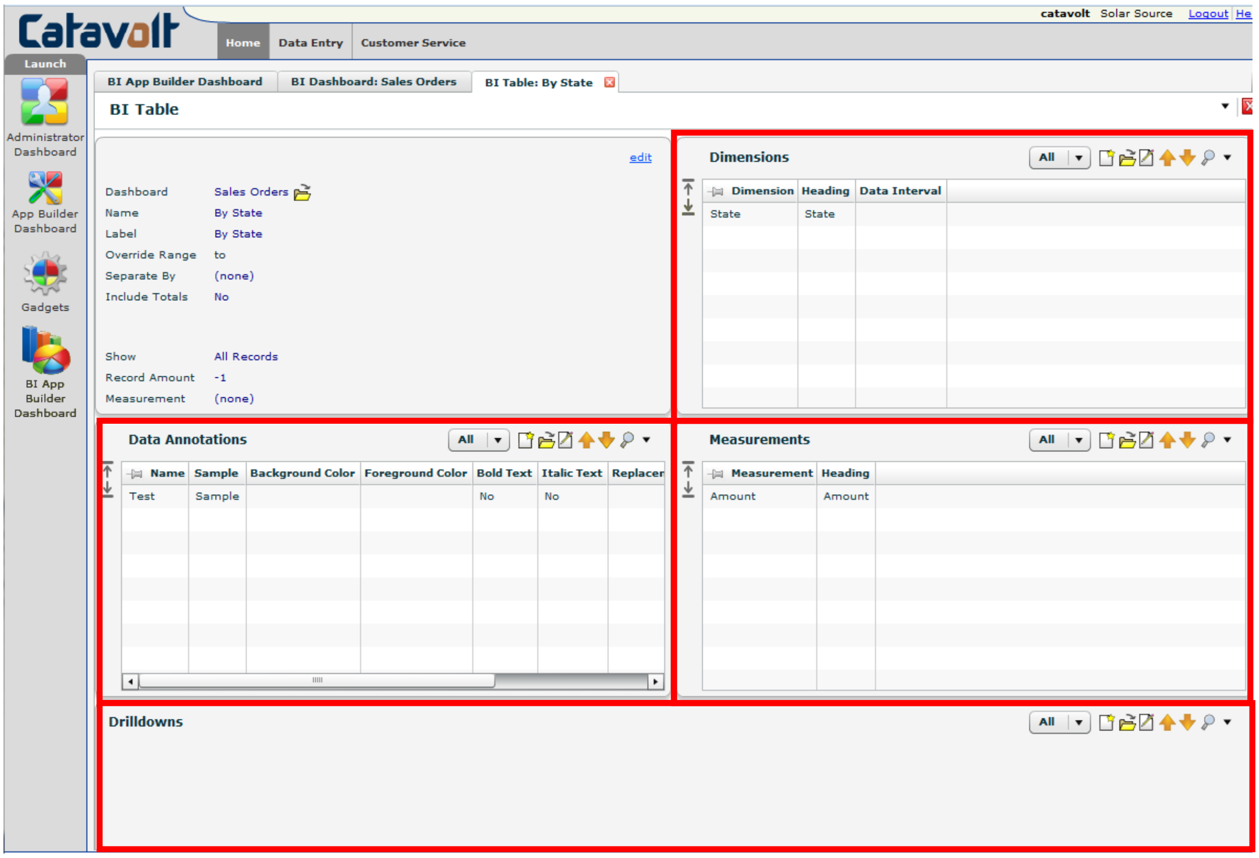
- All Records – Build the table using all records
- Bottom – Build the table using the bottom (lowest value) records
- Top – Build the table using the top (highest value) records

Record Amount specifies the number of records to show when building the table. This value is available when you specify Bottom or Top for the Show field.

Measurement specifies the measurement to use when sorting records for the table. This value is available when you specify Bottom or Top for the Show field. Using Show, Record Amount, and Measurement allows you to display information such as “Show the Top 10 States by Sales Amount” or “Show the Bottom 10 Months by Total Revenue”.

BI Table Components

BI Tables are made up of 4 components: Dimensions (a list of Dimensions to show in the table) Measurements (a list of Measurements to show in the table), Data Annotations (the ability to annotate data using colors, fonts and text) and Drilldowns (references to other Graphs, Tables, etc that a user can further drill into when opening a value on the Table).



BI Table

Dashboard	Sales Orders
Name	By State
Label	By State
Override Range	to
Separate By	(none)
Include Totals	No
Show	All Records
Record Amount	-1
Measurement	(none)

Dimensions

Dimension	Heading	Data Interval
State	State	

Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacer
Test	Sample			No	No	

Measurements

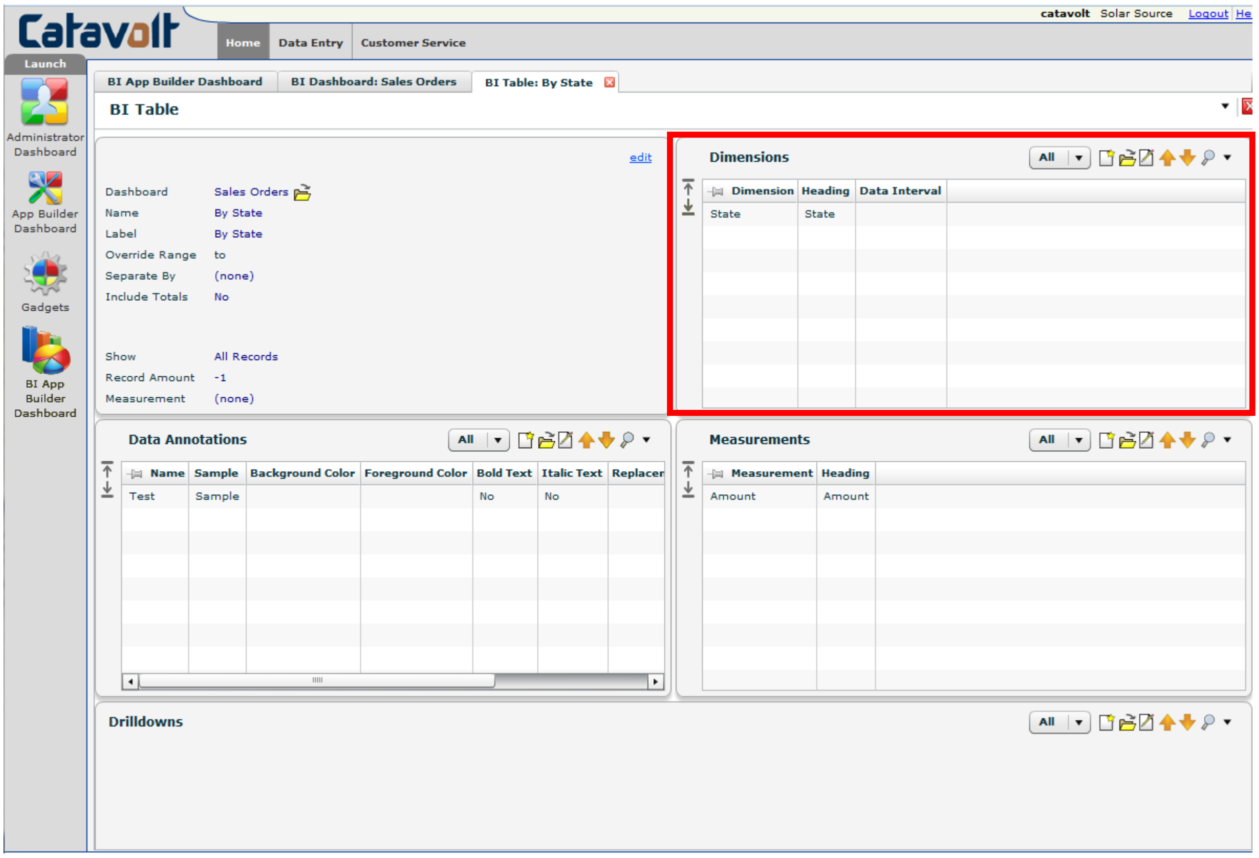
Measurement	Heading
Amount	Amount

Drilldowns

Figure 45: BI Table with Dimensions, Data Annotations, Measurements, and Drilldowns sections highlighted

BI Table Dimensions

BI Table Dimensions specify columns used to break up calculations for measurements. Each combination of values for the specified Dimensions will result in a row in the BI Table. Measurements will be calculated for each combination of Dimension values. Multiple Dimensions can be created for a BI Table. The dimension columns will appear in the BI Table in the same order they appear in the Dimensions section. To change the order, select a Dimension and press the Move Up and Move Down toolbar buttons.



BI Table

Dashboard	Sales Orders
Name	By State
Label	By State
Override Range	to
Separate By	(none)
Include Totals	No
Show	All Records
Record Amount	-1
Measurement	(none)

Dimensions

Dimension	Heading	Data Interval
State	State	

Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacer
Test	Sample			No	No	

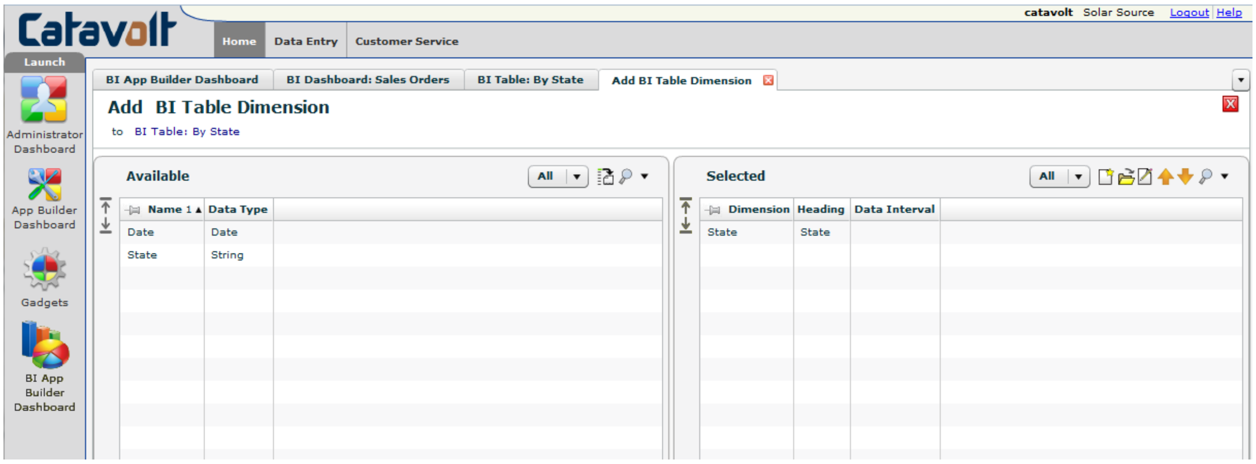
Measurements

Measurement	Heading
Amount	Amount

Drilldowns

Figure 46: BI Table details view with Dimensions section highlighted

When adding BI Table Dimensions, you will be presented with two lists. The Available Dimensions list shows all Dimensions for the Data Set. The Selected Dimensions list shows the dimensions that will be available to the BI Table.



Add BI Table Dimension

to BI Table: By State

Available

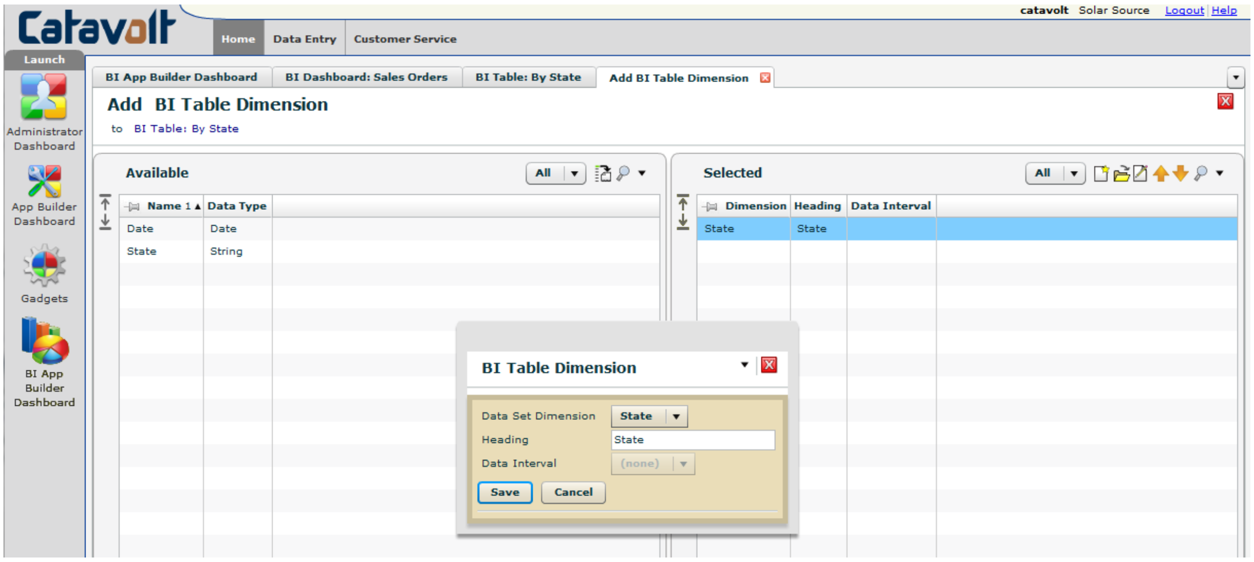
Name 1 ▲	Data Type
Date	Date
State	String

Selected

Dimension	Heading	Data Interval
State	State	

Figure 47: Add BI Table Dimension view

You can select a single or multiple Dimensions and press the Add button to add them to the BI Table. If you select a single dimension, you will be presented with the following dialog:



Add BI Table Dimension

to BI Table: By State

Available

Name 1 ▲	Data Type
Date	Date
State	String

Selected

Dimension	Heading	Data Interval
State	State	

BI Table Dimension

Data Set Dimension	State
Heading	State
Data Interval	(none)

Figure 48: BI Table Dimension create details view

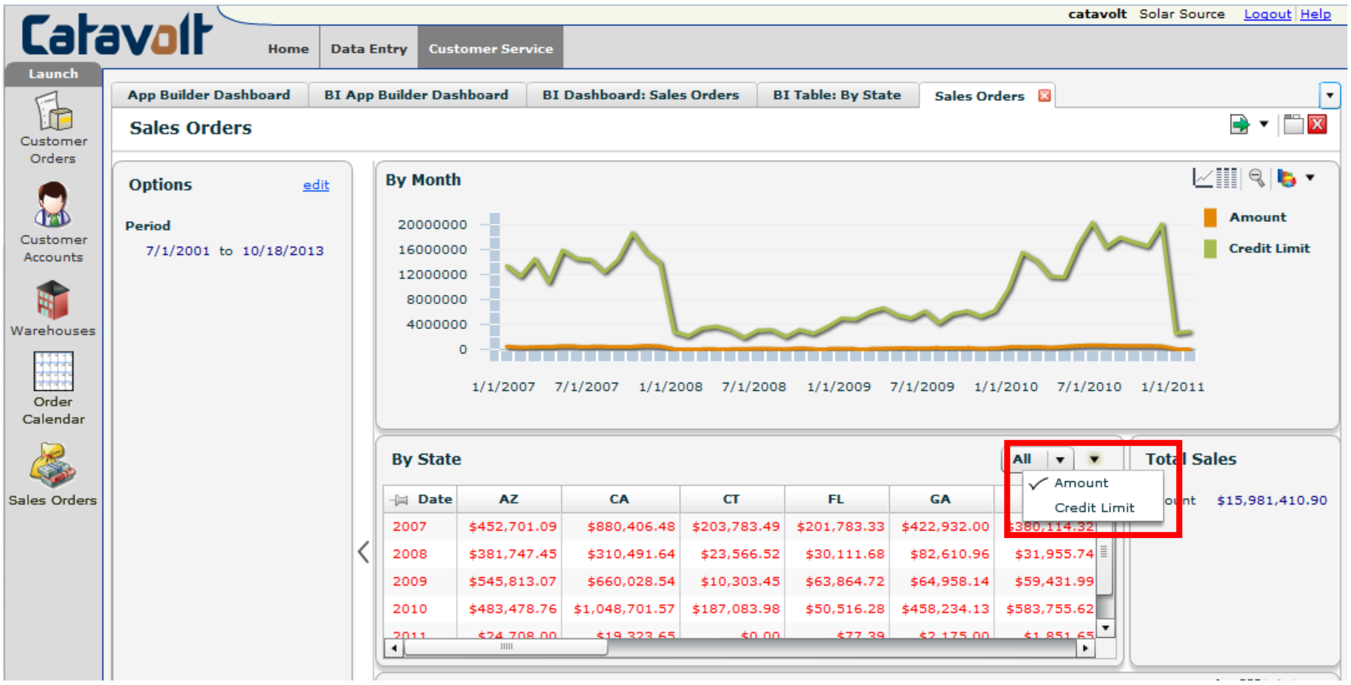
When creating a BI Table Dimension, you must specify a Data Set Dimension. This specifies the dimension that will be used to create a column in the BI Table.

Heading specifies the heading of the column in the BI Table. This defaults to the Dimension name.

Display Interval specifies the time interval grouping for the column of the BI Table. This value is available for Dimensions that are of type Date. The available values are Day, Week, Month, Quarter, and Year.

BI Table Measurements

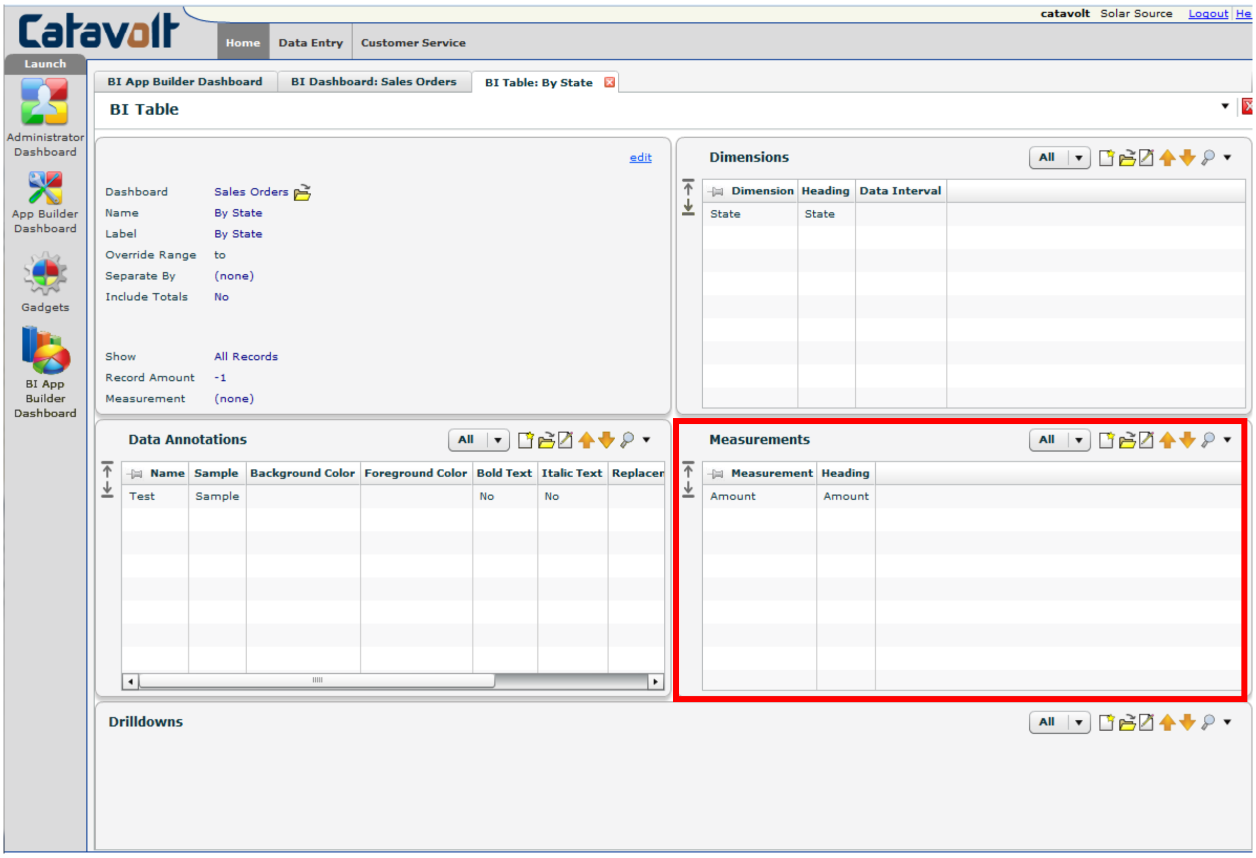
BI Table Measurements specify columns used to calculate values for the BI Table. Measurement values are calculated based on the combination of values for the specified Dimensions. Multiple Measurements can be created for a BI Table. The only exception is if you are showing a Table with a Separate By property. In this case, since the values of the Separate By value are already showing multiple columns, we only show one of the BI Table Measurements on the table itself. You can use the Table's dropdown menu to choose a different measure to show.



Date	AZ	CA	CT	FL	GA	
2007	$452,701.09	$880,406.48	$203,783.49	$201,783.33	$422,932.00	$380,114.32
2008	$381,747.45	$310,491.64	$23,566.52	$30,111.68	$82,610.96	$31,955.74
2009	$545,813.07	$660,028.54	$10,303.45	$63,864.72	$64,958.14	$59,431.99
2010	$483,478.76	$1,048,701.57	$187,083.98	$50,516.28	$458,234.13	$583,755.62
2011	$24,708.00	$19,323.65	$0.00	$77.39	$2,175.00	$1,851.65

Figure 49: BI Table using Separate By property with multiple measurements highlighted

The Measurement columns will appear in the BI Table in the same order they appear in the Measurements section. To change the order, select a Measurement and press the Move Up and Move Down toolbar buttons.



BI Table

Dashboard	Sales Orders
Name	By State
Label	By State
Override Range	to
Separate By	(none)
Include Totals	No
Show	All Records
Record Amount	-1
Measurement	(none)

Dimensions

Dimension	Heading	Data Interval
State	State	

Data Annotations

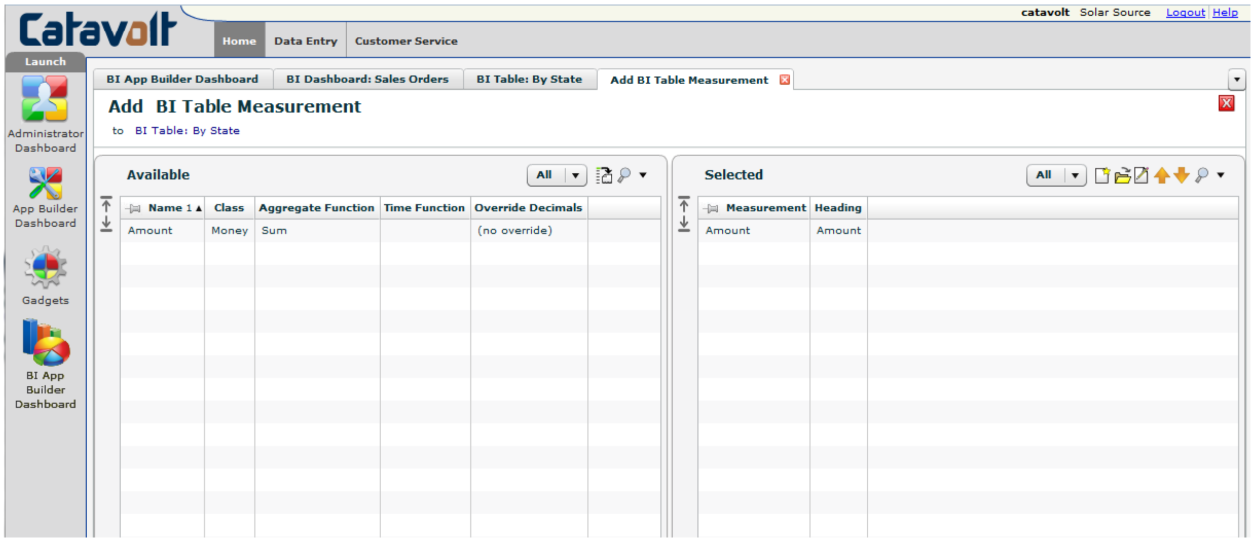
Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacer
Test	Sample			No	No	

Measurements

Measurement	Heading
Amount	Amount

Figure 50: BI Table details view with Measurements section highlighted

When adding BI Table Measurements, you will be presented with two lists. The Available Measurements list shows all Measurements for the Data Set. The Selected Measurements list shows the measurements that will be shown on the BI Table.

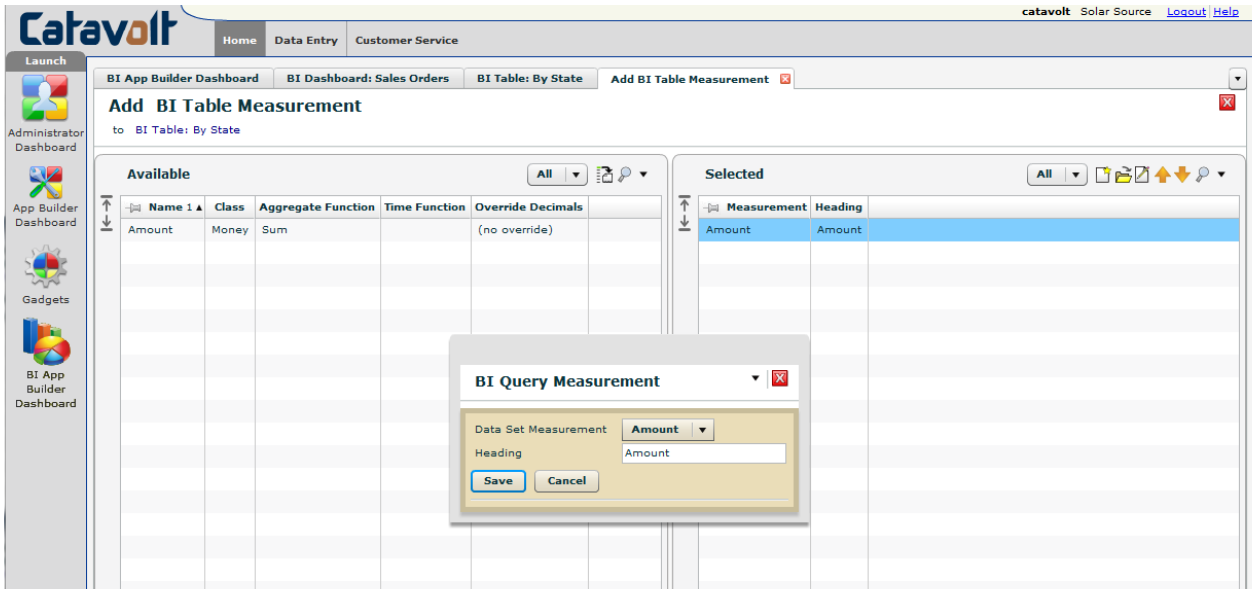


Name 1	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)

Measurement	Heading
Amount	Amount

Figure 51: Add BI Table Measurement view

You can select a single or multiple Measurements and press the Add button to add them to the BI Table. If you select a single measurement, you will be presented with the following dialog:



Name 1	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)

Measurement	Heading
Amount	Amount

Data Set Measurement: Amount
Heading: Amount

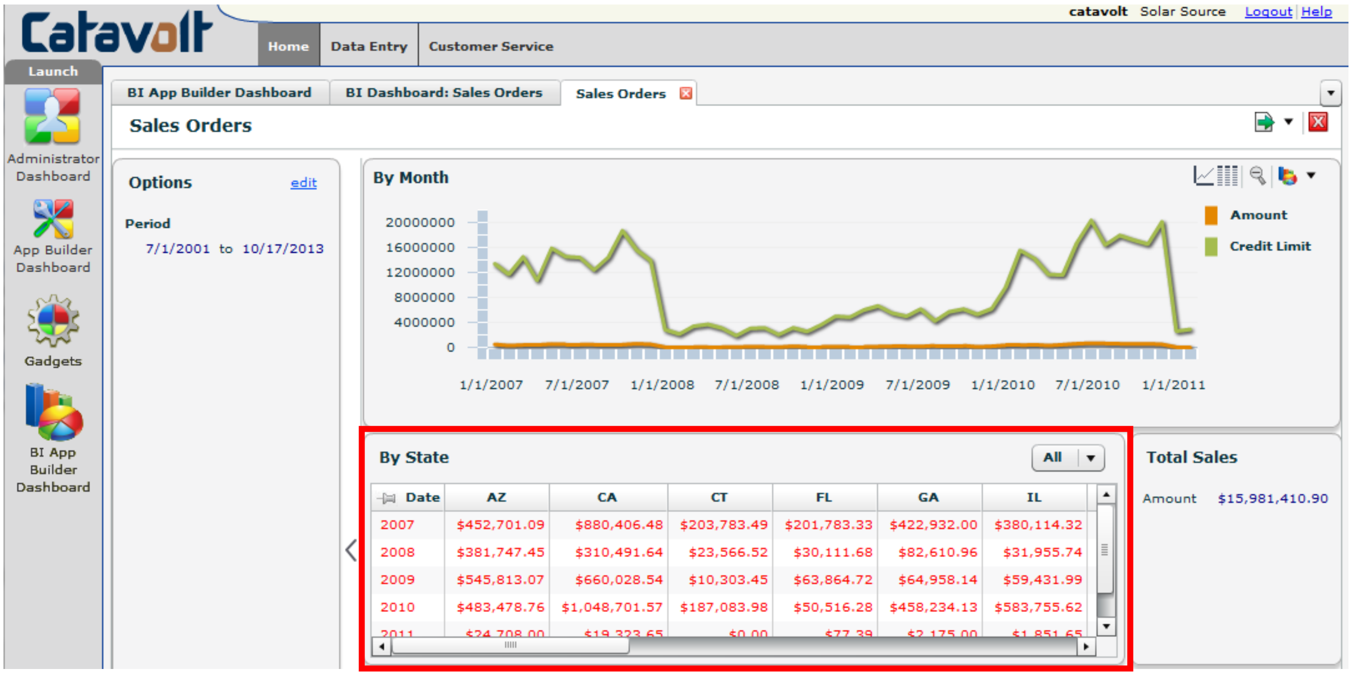
Figure 52: BI Table Measurement create details view

When creating a BI Table Measurement, you must specify a Data Set Measurement. This specifies the measurement that will be used to create a column in the BI Table.

Heading specifies the heading of the column in the BI Table. This defaults to the Measurement name.

BI Table Data Annotations

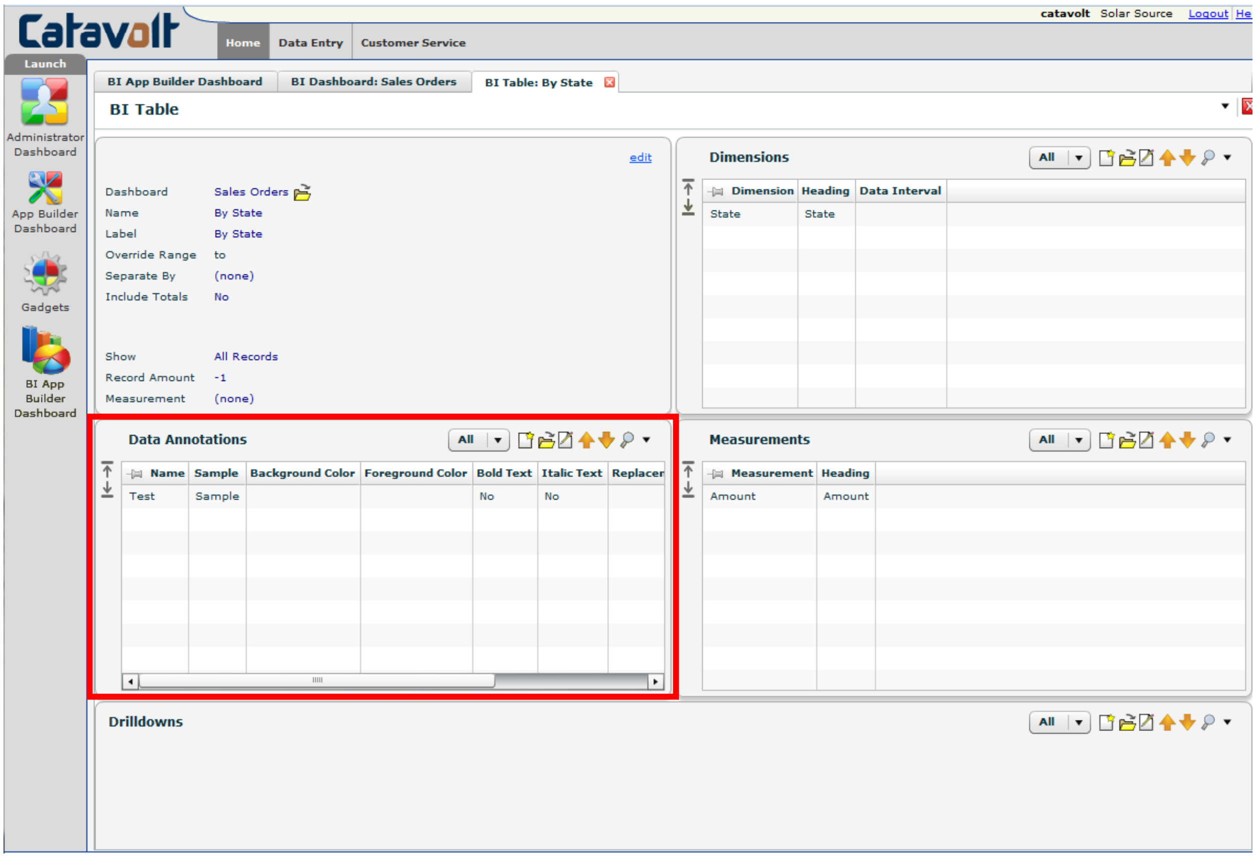
Data Annotations allow you to highlight BI Table data using colors, fonts, images and alternate text. Data Annotations allow you to add background colors, foreground colors, bold lettering, italic lettering and override text (to either entire rows or cells within a row) based on the data being displayed. The end user sees the annotations in their table.



Date	AZ	CA	CT	FL	GA	IL
2007	$452,701.09	$880,406.48	$203,783.49	$201,783.33	$422,932.00	$380,114.32
2008	$381,747.45	$310,491.64	$23,566.52	$30,111.68	$82,610.96	$31,955.74
2009	$545,813.07	$660,028.54	$10,303.45	$63,864.72	$64,958.14	$59,431.99
2010	$483,478.76	$1,048,701.57	$187,083.98	$50,516.28	$458,234.13	$583,755.62
2011	$24,708.00	$19,323.65	$0.00	$77.39	$2,175.00	$1,851.65

Figure 53: BI Dashboard showing a Table with data annotations highlighted

Multiple Data Annotations are allowed for a BI Table. Each Data Annotation can affect either one or more individual columns or the entire row. When multiple Data Annotations affect the same View column or row, the last Annotation in the list will be applied. To change the order, select a Data Annotation and press the Move Up and Move Down toolbar buttons.



BI Table

Dashboard	Sales Orders
Name	By State
Label	By State
Override Range	to
Separate By	(none)
Include Totals	No
Show	All Records
Record Amount	-1
Measurement	(none)

Dimensions

Dimension	Heading	Data Interval
State	State	

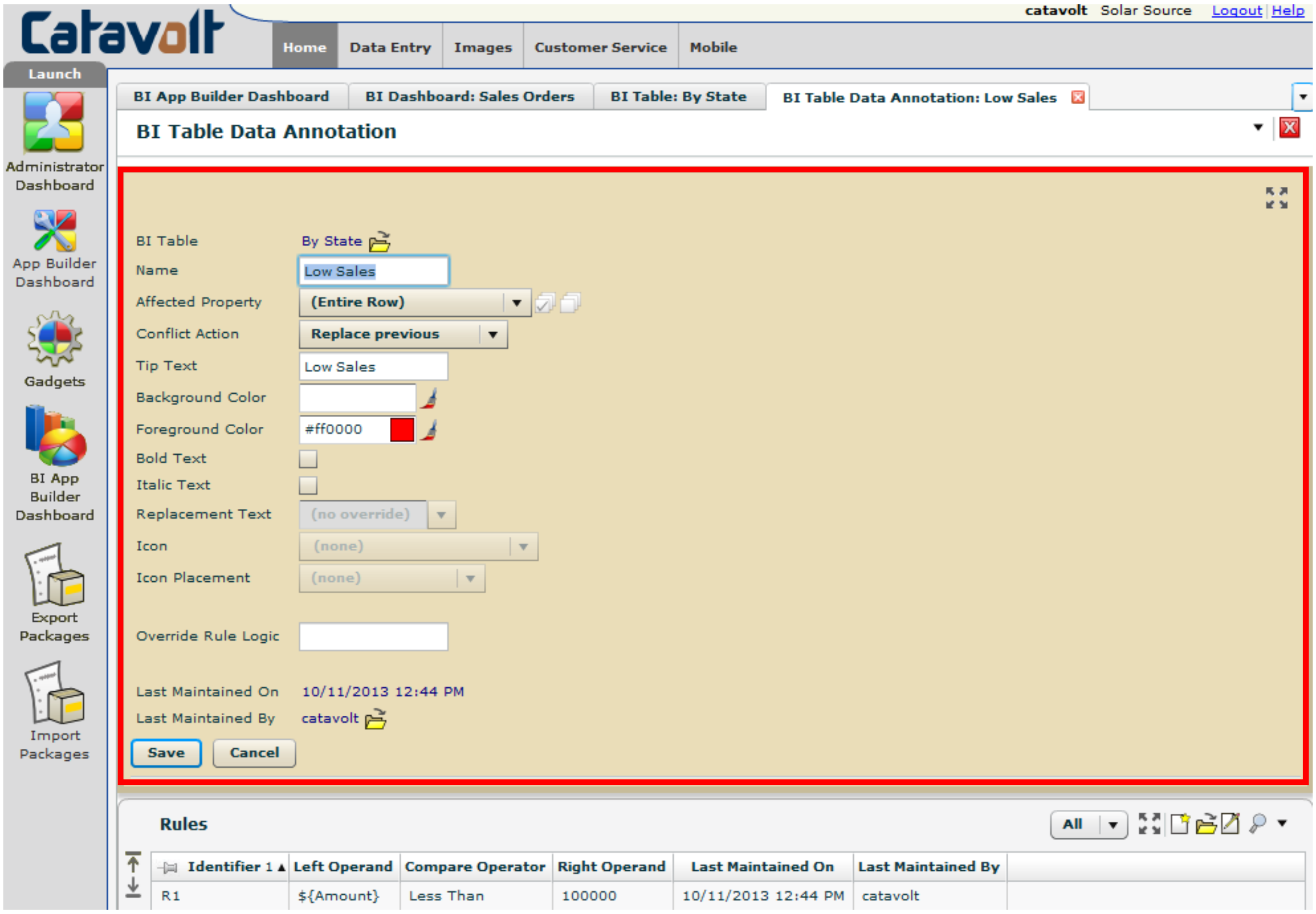
Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacer
Test	Sample			No	No	

Measurements

Measurement	Heading
Amount	Amount

Figure 54: BI Table details view with Data Annotations section highlighted



BI Table Data Annotation

BI Table	By State
Name	Low Sales
Affected Property	(Entire Row)
Conflict Action	Replace previous
Tip Text	Low Sales
Background Color	
Foreground Color	#ff0000 🟥
Bold Text	□
Italic Text	□
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)
Override Rule Logic	
Last Maintained On	10/11/2013 12:44 PM
Last Maintained By	catavolt

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand	Last Maintained On	Last Maintained By
R1	${Amount}	Less Than	100000	10/11/2013 12:44 PM	catavolt

Figure 55: BI Table Data Annotation create details view

When creating a Data Annotation, you must specify a Name. This value is used to uniquely identify the Data Annotation and is not presented to the end user.

Affected Property specifies which View Property(s) will be altered by the Data Annotation or if it applies to the whole record. Xalt defaults this value to (Entire Row), which will alter every column for the specified row. You also have the option of choosing one or more columns from the Table.

Conflict Action specifies what should be done if multiple Data Annotations apply to a single row or a single column. As stated above, Annotations are processed in a specific sequence. Two annotations are in conflict with each other when they both pass all their rules for a row and are annotating the same column or the entire row. When annotations are in conflict, the last one in the list wins. This option allows the annotation to either be merged with previous annotations or replace previous annotations. When an annotation is merged with a previous annotation, it will assume all previous annotations' properties that it does not specify itself. For example, if one annotation sets the background color and a latter one specifies a foreground color and they are merged, the resulting annotation will have both a background and a foreground color set.

Tip Text specifies text that will be displayed when the end user hovers over a row or column currently affected by a Data Annotation. In this way, the end user can get a text description of the Data Annotation that is currently being applied to the row or column without having to remember what the different colors, fonts, etc. mean when looking at a Query.

Background Color specifies the color you wish to change the background of the affected cell or the entire row if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Foreground Color specifies the color you wish to change the foreground text of the affected cell or the entire row if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Bold Text specifies whether you want to change the font to be bold. Select the checkbox to apply a bold font to the affected cell or the entire row if the Data Annotation is applied.

Italic Text specifies whether you want to change the font to be italicized. Select the checkbox to apply an italicized font to the affected cell or the entire row if the Data Annotation is applied.

Replacement Text specifies whether you want to replace the value of the affected cell with an alternate text string. This is commonly used to improve readability of the Table (for example, to replace 0 values with blanks). Note that Replacement Text is not available when the Affected Property is (Entire Row).

Icon specifies whether you want to append or replace the value of the affected cell with an image. The image you choose can be shown instead of or in addition to the value of the affected cell. Note that Icon is not available when the Affected Property is (Entire Row).

Icon Placement specifies where to place the image in relation to the value of the affected cell. The available values are:

- Replace Text – Replace the value of the affected cell with the image.
- Left of Text – Put the image to the left of the value of the affected cell.
- Right of Text – Put the image to the right of the value of the affected cell.
- Background – Put the image in the background under the value of the affected cell. This value is not used on mobile clients.
- Background (Fill) – Put the image in the background under the value of the affected cell and stretch the image to fill the column horizontally. This is typically used for annotations that show progress bars. This value is not used on mobile clients

Override Rule Logic specifies more advanced logic to evaluate the Data Annotation Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Data Annotation to be applied. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

(R1 AND R2) OR (R3 AND R4)

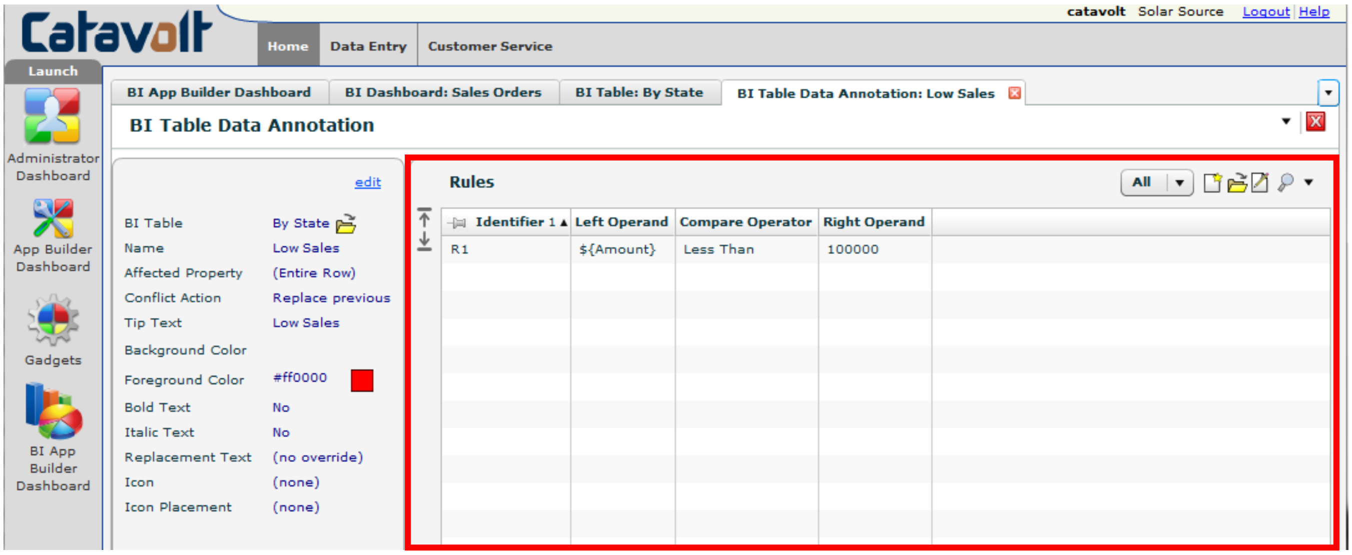
NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

BI Table Data Annotation Rules

The Rules list on a Data Annotation can be used to control when the Data Annotation should be applied to a row of data. A Data Annotation can have zero or more rules. All rules must pass for the Data Annotation to be applied to that particular row of data. If a Data Annotation has no rules, then it is considered to pass automatically.



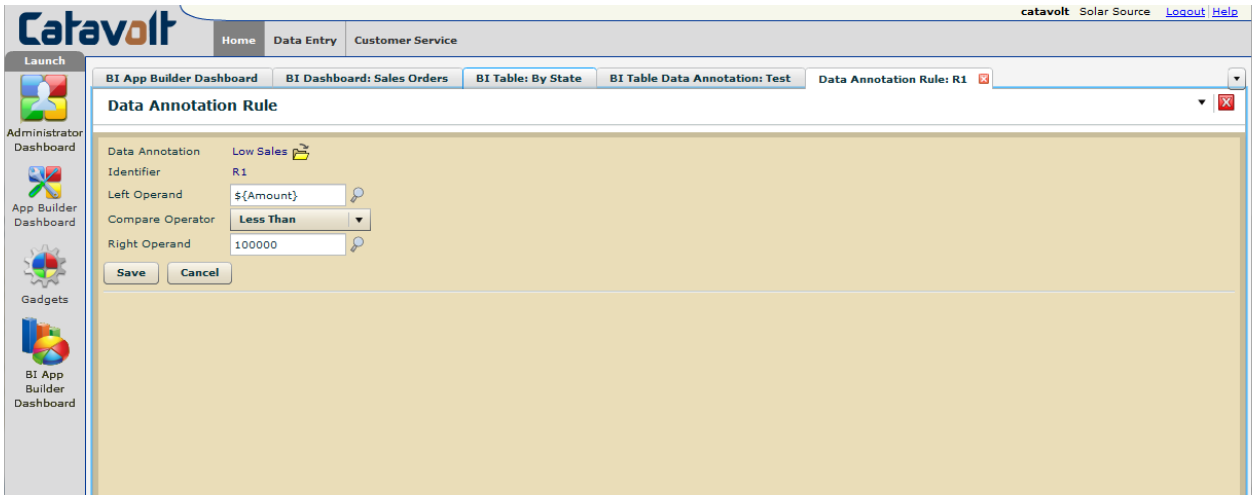
BI Table Data Annotation

BI Table	By State
Name	Low Sales
Affected Property	(Entire Row)
Conflict Action	Replace previous
Tip Text	Low Sales
Background Color	
Foreground Color	#ff0000 🟥
Bold Text	No
Italic Text	No
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand
R1	${Amount}	Less Than	100000

Figure 56: BI Table Data Annotation details view with Rules section highlighted



Data Annotation Rule

Data Annotation	Low Sales
Identifier	R1
Left Operand	${Amount}
Compare Operator	Less Than
Right Operand	100000

Figure 57: BI Table Data Annotation create details view

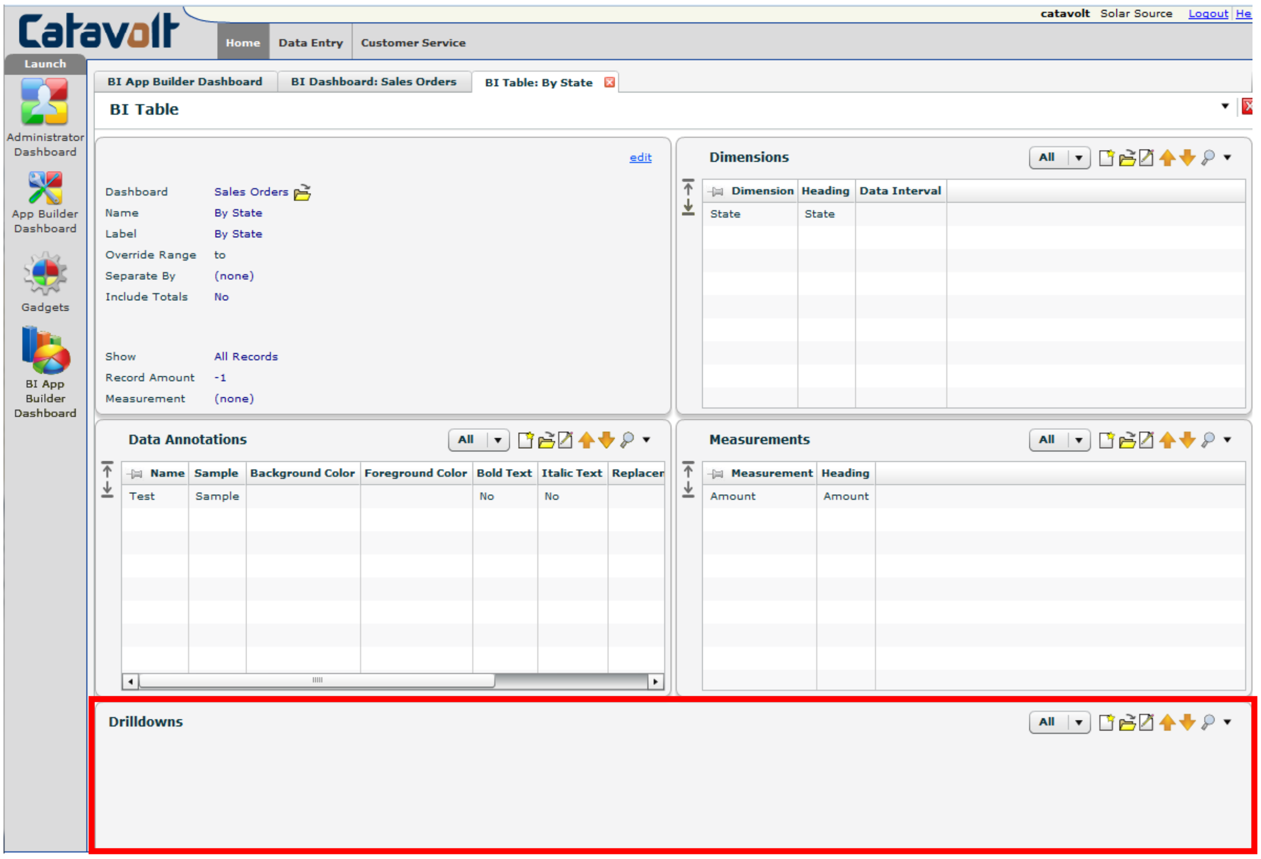
When creating a Data Annotation Rule, Xalt will automatically create a unique Identifier for the Rule.

Left Operand and Right Operand specify the two values that will be compared with each other. You have 2 choices for operands: Constant Values (such as A or 1.00) or Substitution Values. You can enter a Substitution Value manually or press the Find button to have Xalt build a Substitution Value based on the Available Properties for the Data Object. Appendix A: Specifying Messages and Substitution Values for more information about how to create Substitution Messages for Left Operand and Right Operand. A special Substitution Value of ${AFFECTED_PROPERTY} is available for annotations where you select multiple Affected Properties. When running the annotation, the rule will be checked independently for each affected property. This allows you to create a single Annotation and a single rule that can apply to multiple individual properties. For example, if you have 5 numeric columns in a Table and you want any negative values to show as red, you can create a single annotation to make the text foreground color red, choose all 5 numeric columns as Affected Properties, and create a single Annotation Rule of ${AFFECTED_PROPERTY} Less Than 0.

Compare Operator specifies how the two operands should be compared with each other. Options include Equal, Not Equal, Greater Than, Less Than, Greater or Equal, Less or Equal, Contains, Starts With, Ends With, Is Null, and Is Not Null.

BI Table Drilldowns

BI Table Drilldowns provides a way users can go from the current BI Table and "drill-down" into another BI Dashboard, a URL, etc. Multiple Drilldowns can be created for a BI Table. They will appear in the Table menu in the same order they appear in the Drilldowns section. To change the order, select a Drilldown and press the Move Up and Move Down toolbar buttons.



Dashboard Sales Orders
Name By State
Label By State
Override Range to
Separate By (none)
Include Totals No

Show All Records
Record Amount -1
Measurement (none)

Dimensions

Dimension	Heading	Data Interval
State	State	

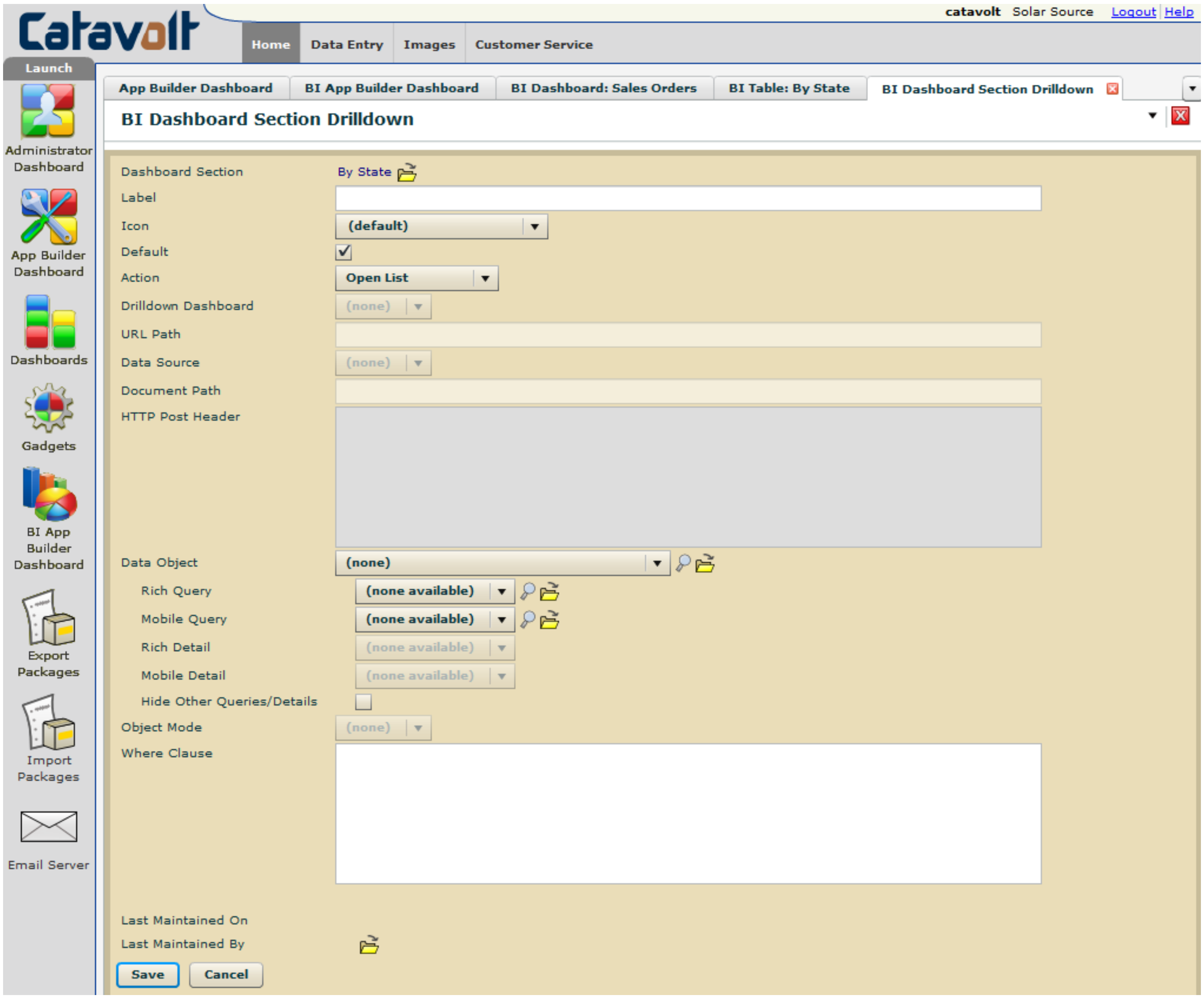
Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacer
Test	Sample			No	No	

Measurements

Measurement	Heading
Amount	Amount

Figure 58: BI Table details view with the Drilldowns section highlighted



BI Dashboard Section Drilldown

Dashboard Section	By State
Label	
Icon	(default)
Default	☑
Action	Open List
Drilldown Dashboard	(none)
URL Path	
Data Source	(none)
Document Path	
HTTP Post Header	
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)
Rich Detail	(none available)
Mobile Detail	(none available)
Hide Other Queries/Details	☐
Object Mode	(none)
Where Clause	
Last Maintained On	
Last Maintained By	

Figure 59: BI Table Drilldown create details view

When creating a BI Table Drilldown, you must specify a Label. This specifies the name of the drilldown menu action that the table will use.

Icon specifies the image that will be used when displaying the action on the context menu and toolbar. You can choose the default Hexagon image or any other image you have uploaded (To upload an image, choose the Upload Image menu action from the Data Sources list).

Default specifies that this is the "default" drilldown action, that is, the action that will be performed when you double-click a table record. Drilldowns that are not the default must be selected via the table's context menu and toolbar. You may only have one Drilldown that has been selected as default per BI Table.

Action specifies the action that will be taken when the Drilldown's menu action is selected. The available options are:

- Open Dashboard – Open another BI Dashboard

- Open Document – Open Document will download a specified document via FTP and display it in the client (if Allow FTP is set to Yes for the Action's Data Source). You may also specify a private URL that will be accessed via the Connector gateway and then downloaded and displayed by the client
- Open URL – Open URL will open a new browser tab and display the specified URL.
- Open List – Open List will open a new tab and display a Data Object Query.
- Open Object – Open Object will open a new tab and display a Data Object Detail in either read or update mode.

Drilldown Dashboard specifies the BI Dashboard that will be displayed if your Action is Open Dashboard. Any BI Dashboard Filters in common between the BI Dashboards will be passed down to the Drilldown Dashboard from the current BI Dashboard.

URL Path specifies the public URL that will be displayed if your Action is Open URL.

Data Source specifies the Data Source that will be used to FTP the document to display if your Action is Open Document. Note that FTP must be enabled on the Data Source you select.

Document Path allows you to specify the path and name of the document that is to be downloaded if your Action is Open Document. The value of this field will be appended to the FTP Path field in the Data Source in order to get the full path to be downloaded via FTP from the back end system. If a URL is specified for Open Document, the target of the URL (an image, PDF, etc.) will be downloaded from the Connector gateway and sent to the client.

HTTP Post Header allows you to open a URL via an HTTP Post instead of an HTTP Get. You can specify parameters in the HTTP Post Header to use when making the HTTP Post call.

Data Object specifies which Data Object you want to display either a list or object of or perform an action against. This value is only allowed if the Action is Open Object or Open List.

Rich Query specifies which Query you want to display on the list when using a rich client. This value is only allowed if the Action is Open List.

Mobile Query specifies which Query you want to display on the list when using a mobile client. This value is only allowed if the Action is Open List.

Rich Detail specifies which Detail you want to display on the object when using a rich client. This value is only allowed if the Action is Open Object.

Mobile Detail specifies which Detail you want to display on the object when using a mobile client. This value is only allowed if the Action is Open Object.

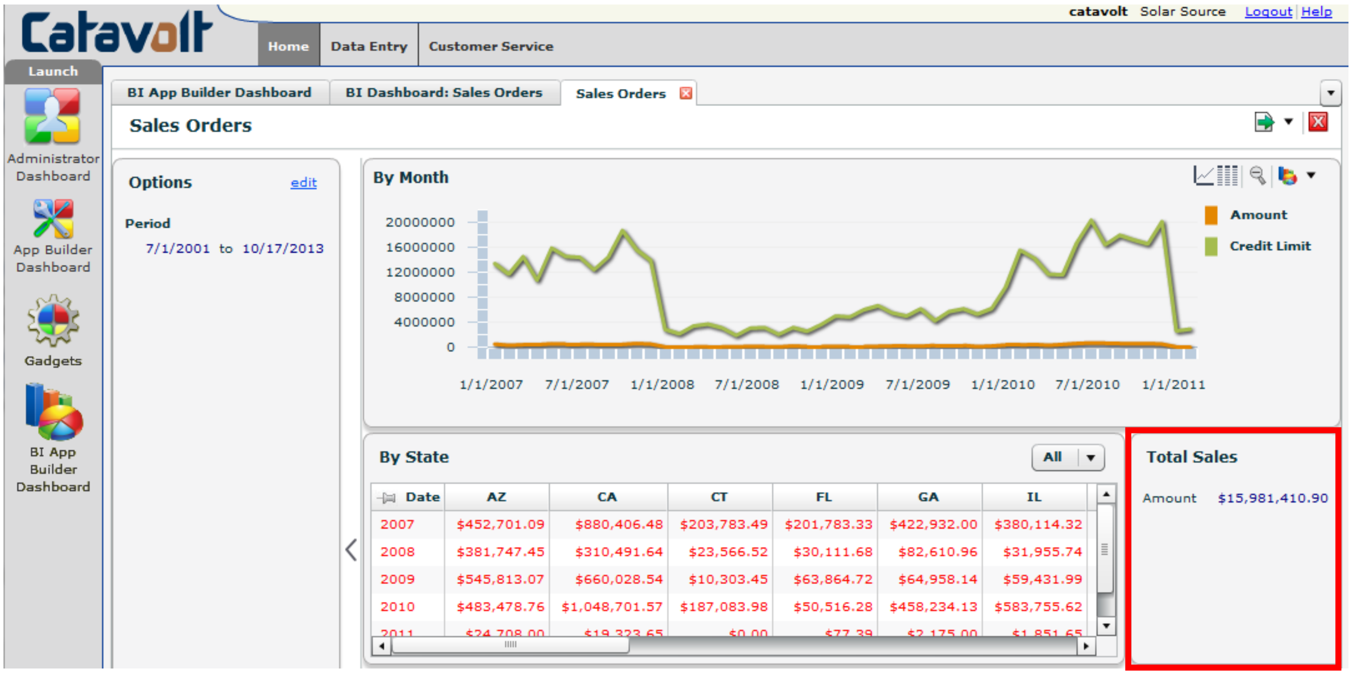
Hide Other Queries/Details specifies whether the runtime Query/Detail dropdown list will contain just the selected Rich/Mobile Query/Detail or all non-hidden Queries/Details. If Hide Other Queries/Details is selected, only the selected Rich/Mobile Query/Detail will be available to choose in the Query/Detail dropdown list.

Object Mode specifies to display the Detail in read-only or update mode when displaying an object. This value is only allowed if the Action is Open Object.

Where Clause specifies extra criteria to use when querying an object. If the Action is Open Object, then Where Clause must be written to return a single record when it is performed (same as when creating a Dashboard Launcher for a Data Object). If the Action is Open List, Where Clause can be optionally specified to further restrict the list of records returned. This value is only allowed if the Action is Open Object or Open List. You can access Dimensions in the Table's Dimension list in the Where Clause. For example, if you have Dimensions of "Date", "State", and "City", you can use ${Date}, ${State}, and ${City} in the Where Clause. Note that if you have a Dimension with a Data Interval other than Day, you may need to manipulate the values passed to you. For example, if you have a Display Interval of Month and a Postgresql database, your Where Clause would be the following in order to see all records for the specified month: shipto_city = '${City}' AND shipto_state = '${State}' AND EXTRACT(YEAR FROM order_date) = EXTRACT(YEAR FROM DATE '${Date}') AND EXTRACT(MONTH FROM order_date) = EXTRACT(MONTH FROM DATE '${Date}')

BI Details

BI Details allows you to present totals data from your Data Set in detail format. A BI Detail consists of one or more measurements. The measurements will be calculated based on all ranges and filters selected from the Options pane.



Sales Orders

Options edit

Period
7/1/2001 to 10/17/2013

By State

Date	AZ	CA	CT	FL	GA	IL
2007	$452,701.09	$880,406.48	$203,783.49	$201,783.33	$422,932.00	$380,114.32
2008	$381,747.45	$310,491.64	$23,566.52	$30,111.68	$82,610.96	$31,955.74
2009	$545,813.07	$660,028.54	$10,303.45	$63,864.72	$64,958.14	$59,431.99
2010	$483,478.76	$1,048,701.57	$187,083.98	$50,516.28	$458,234.13	$583,755.62
2011	$24,708.00	$19,323.65	$0.00	$77.39	$2,175.00	$1,851.65

Total Sales

Amount $15,981,410.90

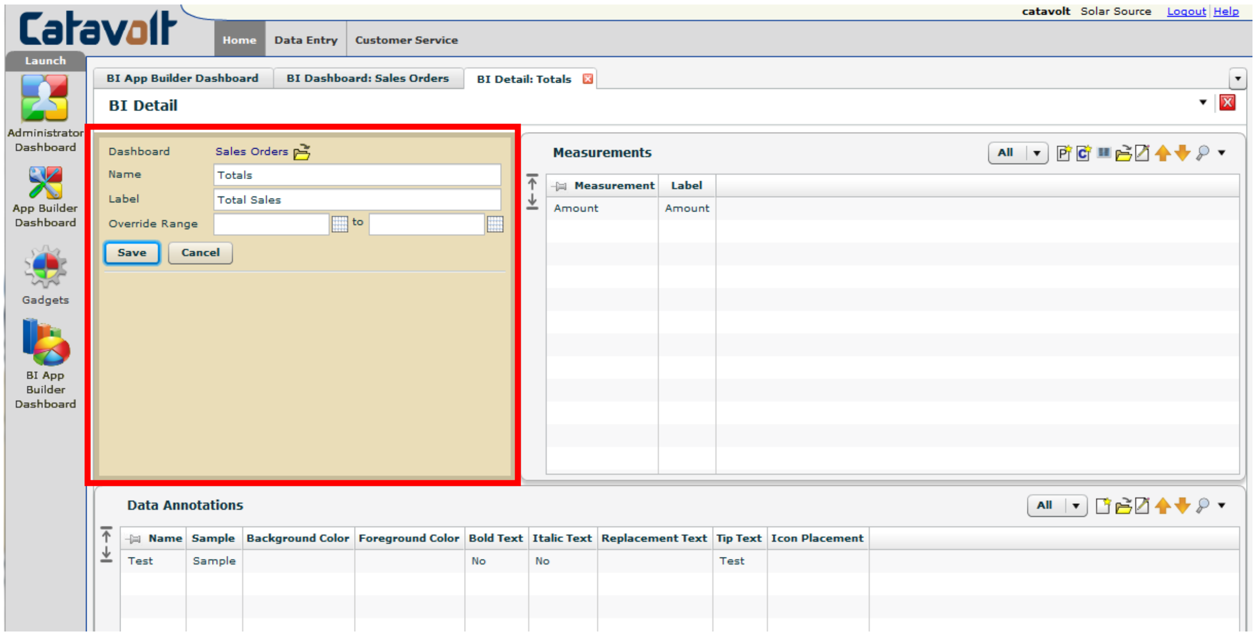
Figure 60: BI Dashboard with a Details section highlighted

Accessing BI Details

To access the Details defined for a particular BI Dashboard, open the BI Dashboard and look Sections section. Multiple Details can be created for a BI Dashboard.

Creating BI Details

To create a BI Detail, choose the Add Detail menu option in the Sections section.



Dashboard	Sales Orders
Name	Totals
Label	Total Sales
Override Range	to

Measurement	Label
Amount	Amount

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacement Text	Tip Text	Icon Placement
Test	Sample			No	No		Test	

Figure 61: BI Details create details view

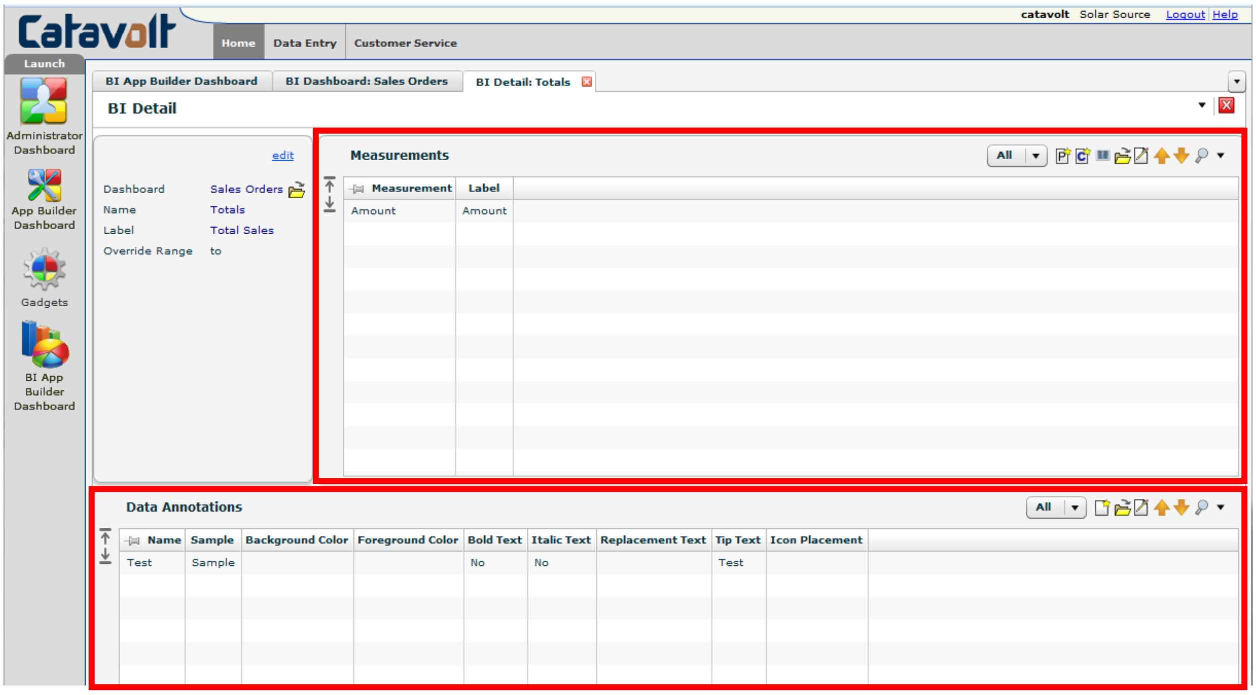
Name specifies the name of the Detail. This value is used to uniquely identify the Detail to the BI Dashboard and will not be displayed to the user.

Label specifies the name that the end user will see when the Detail is displayed.

Override Range allows you to use a different range for this Detail than the range being used for the BI Dashboard. Leave this value blank to use the same range as the BI Dashboard.

BI Detail Components

BI Details are made up of 2 components: Measurements (a list of Measurements to show in the table), and Data Annotations (the ability to annotate data using colors, fonts and text).



Dashboard	Sales Orders
Name	Totals
Label	Total Sales
Override Range	to

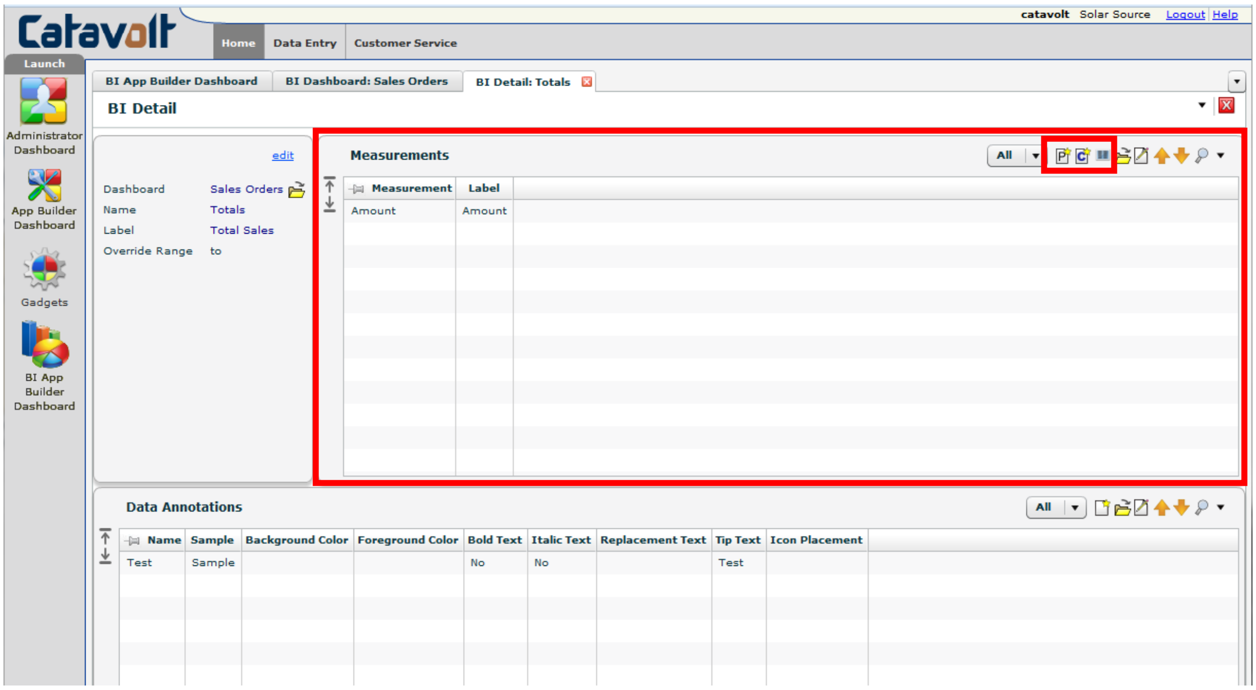
Measurement	Label
Amount	Amount

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacement Text	Tip Text	Icon Placement
Test	Sample			No	No		Test	

Figure 62: BI Detail details view with the Measurements and Data Annotations section highlighted

BI Detail Measurements

BI Detail Measurements specify values used to calculate totals for the BI Detail. Measurement values are calculated based on the ranges and filters selected on the Options pane. You can add Measurements to the BI Detail using the Add Measurement action. You can add constant labels to the BI Detail using the Add Constants action. You can add column breaks using the Add Column action. The Measurement properties will appear in the BI Detail in the same order they appear in the Measurements section. To change the order, select a Measurement and press the Move Up and Move Down toolbar buttons.



BI Detail

Dashboard Sales Orders
Name Totals
Label Total Sales
Override Range to

Measurements

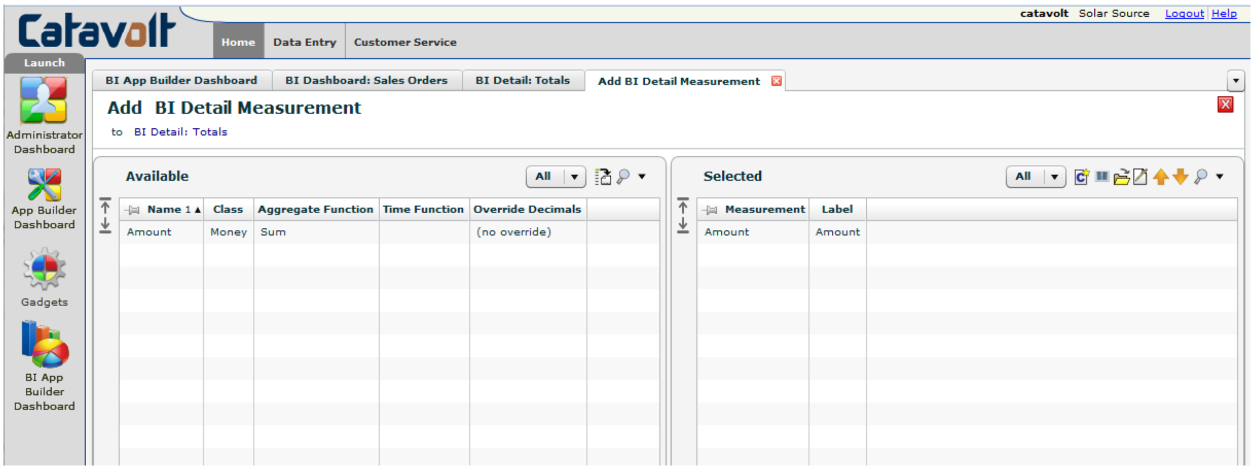
Measurement	Label
Amount	Amount

Data Annotations

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacement Text	Tip Text	Icon Placement
Test	Sample			No	No		Test	

Figure 63: BI Detail details view with the Measurements section highlighted

When adding BI Detail Measurements, you will be presented with two lists. The Available Measurements list shows all Measurements for the Data Set. The Selected Measurements list shows the measurements that will be shown in the BI Detail.



Available

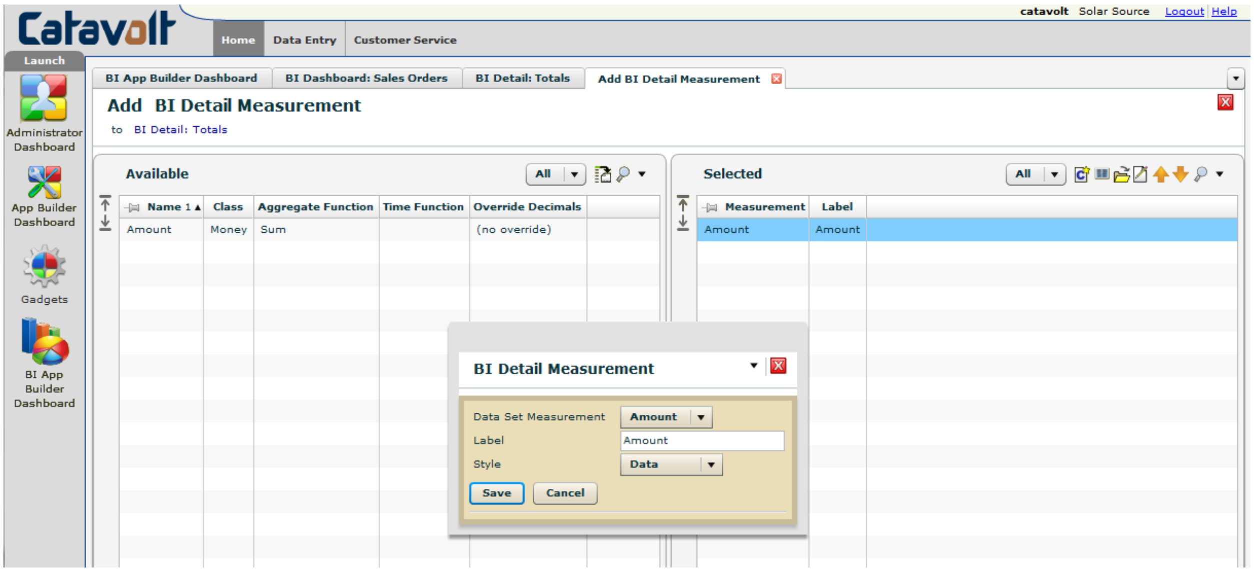
Name 1 ▲	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)

Selected

Measurement	Label
Amount	Amount

Figure 64: Add BI Detail Measurement view

You can select a single or multiple Measurements and press the Add button to add them to the BI Detail. If you select a single measurement, you will be presented with the following dialog:



Available

Name 1 ▲	Class	Aggregate Function	Time Function	Override Decimals
Amount	Money	Sum		(no override)

Selected

Measurement	Label
Amount	Amount

BI Detail Measurement

Data Set Measurement	Amount
Label	Amount
Style	Data

Figure 65: BI Detail Measurement details create view

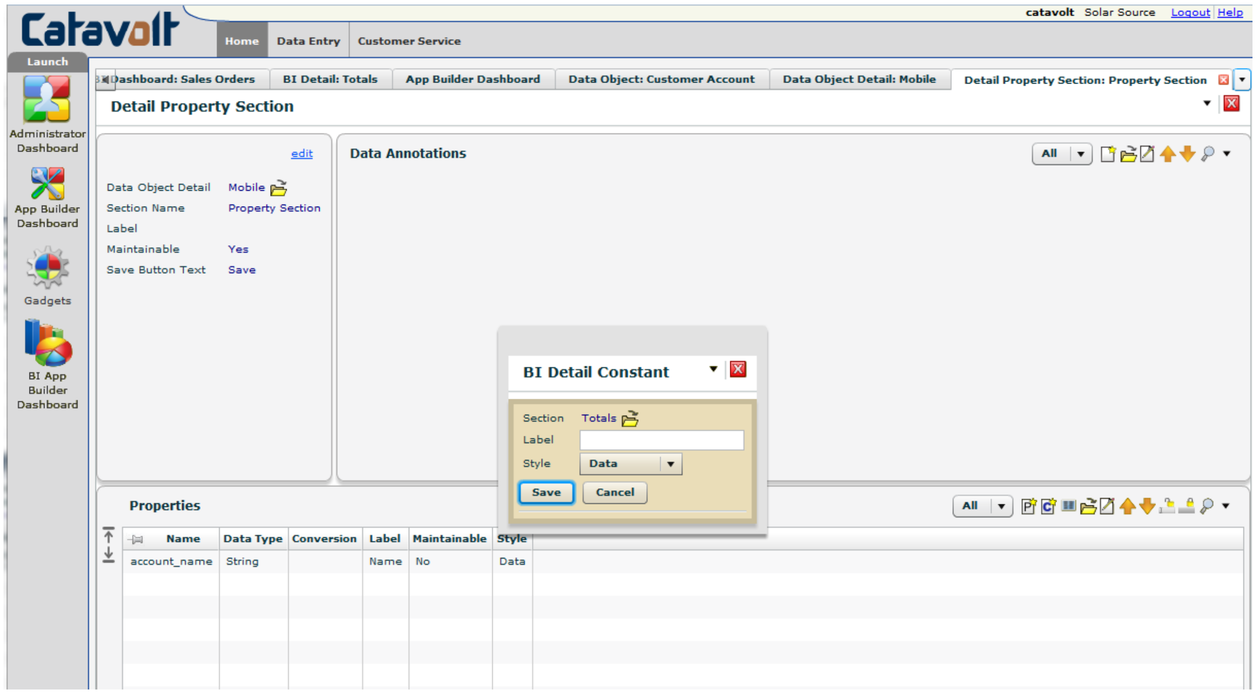
When creating a BI Detail Measurement, you must specify a Data Set Measurement. This specifies the measurement that will be used to create a property in the BI Detail.

Label specifies the label of the property in the BI Detail. This defaults to the Measurement name.

Style specifies the font that is used for displaying this value. Data presents it in a normal font, while Heading 1 through Heading 4 will display the value in bold with different font sizes (Heading 1 being the largest font).

BI Detail Constant Properties

Like their name implies, Constant Properties are text constants that can be used to further group data as it appears in the BI Detail.



Data Object Detail	Mobile
Section Name	Property Section
Label	
Maintainable	Yes
Save Button Text	Save

BI Detail Constant

Section	Totals
Label	
Style	Data

Properties

Name	Data Type	Conversion	Label	Maintainable	Style
account_name	String		Name	No	Data

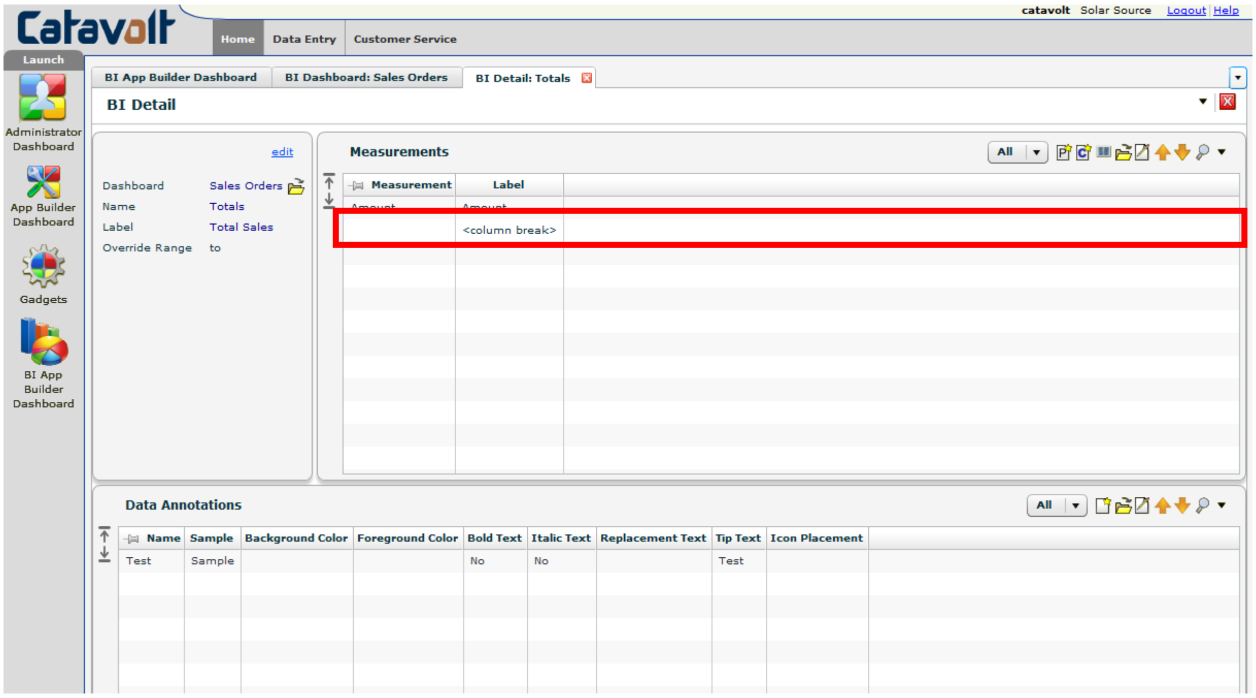
Figure 66: BI Detail Constant Property create details view

Label specifies the description that will be displayed in the section. You can leave this value empty in order to add a blank line to the section.

Style specifies the font that is used for displaying this value. Data presents it in a normal font, while Heading 1 through Heading 4 will display the value in bold with different font sizes (Heading 1 being the largest font).

BI Detail Column Breaks

Column Breaks allow you to display data across multiple columns within a BI Detail.



BI Detail

edit

Dashboard Sales Orders
Name Totals
Label Total Sales
Override Range to

Measurements

Measurement	Label
Amount	Amount
	<column break>

Data Annotations

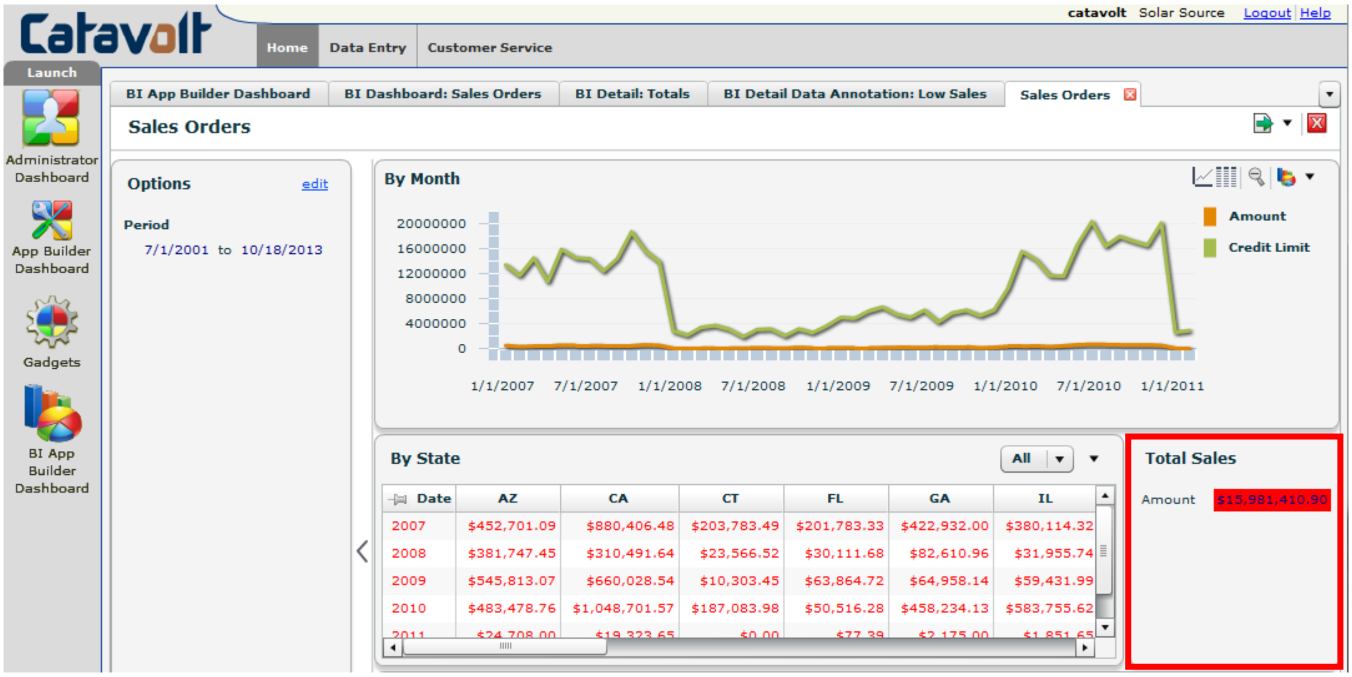
Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacement Text	Tip Text	Icon Placement
Test	Sample			No	No		Test	

Figure 67: BI Detail details view with the column break value highlighted

Selecting the Add Column menu action will add a <column break> indicator to the list of Properties. Everything above the <column break> indicator will be displayed in Column 1, while everything below the indicator will be displayed in Column 2. Multiple column breaks can be added to create more columns. In order for the data to line up correctly, you will need to ensure that columns to the left contain the same number of fields or more than columns to the right.

BI Detail Data Annotations

Data Annotations for BI Details are very similar to Data Annotations for details in that they allow you to highlight Detail data using colors, fonts, images and alternate text. Data Annotations allow you to add background colors, foreground colors, bold lettering, italic lettering and override text based on the data being displayed. The end user sees the annotations in their BI Detail.



Period
7/1/2001 to 10/18/2013

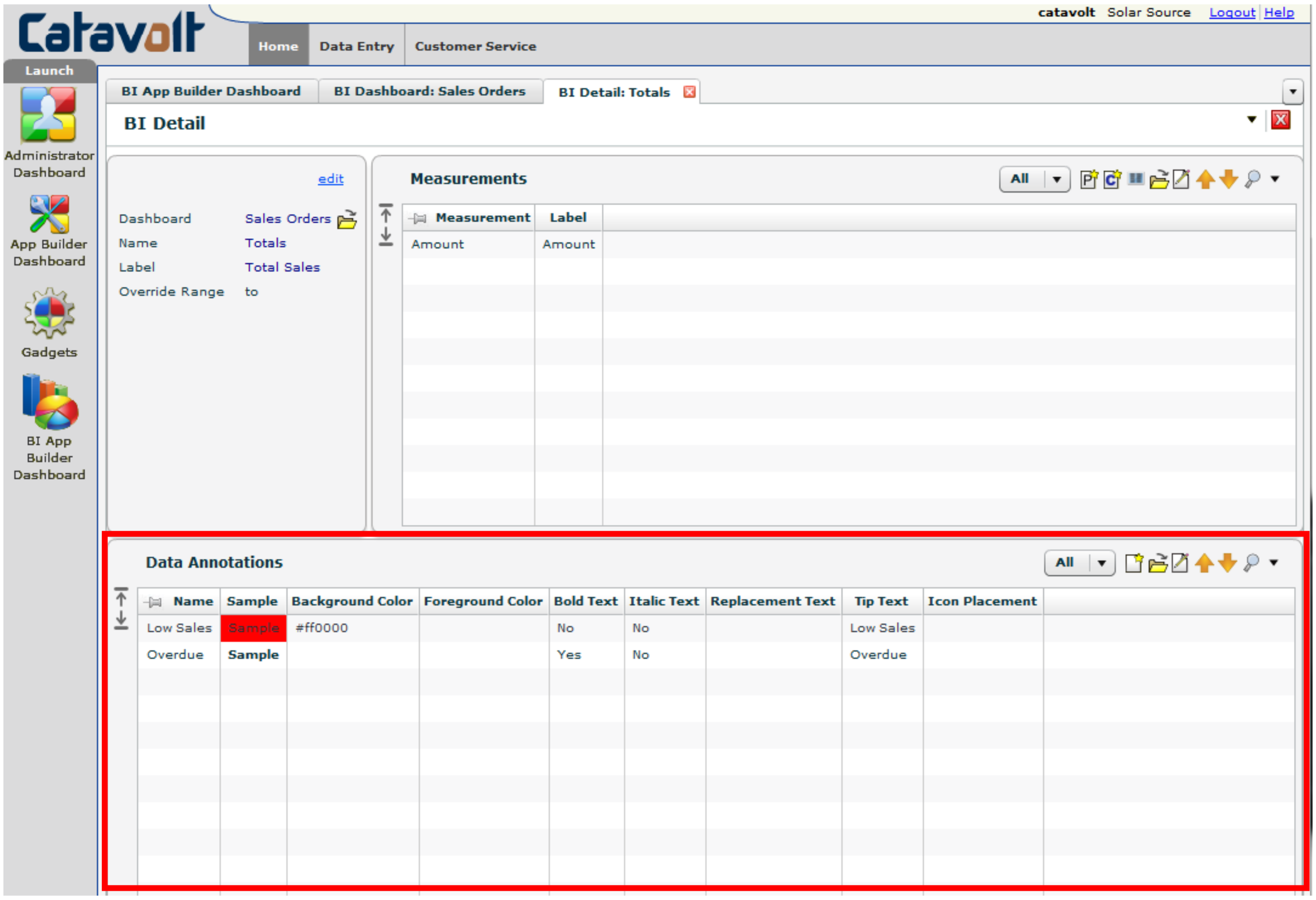
Date	AZ	CA	CT	FL	GA	IL
2007	$452,701.09	$880,406.48	$203,783.49	$201,783.33	$422,932.00	$380,114.32
2008	$381,747.45	$310,491.64	$23,566.52	$30,111.68	$82,610.96	$31,955.74
2009	$545,813.07	$660,028.54	$10,303.45	$63,864.72	$64,958.14	$59,431.99
2010	$483,478.76	$1,048,701.57	$187,083.98	$50,516.28	$458,234.13	$583,755.62
2011	$24,708.00	$19,323.65	$0.00	$77.39	$2,175.00	$1,851.65

Total Sales

Amount $15,981,410.90

Figure 68: BI Dashboard with a Details containing Data Annotations highlighted

Multiple Data Annotations are allowed for a BI Detail. Each Data Annotation can affect either one or more individual Properties or the entire section. When multiple Data Annotations affect the same Property or section, the last Annotation in the list will be applied. To change the order, select a Data Annotation and press the Move Up and Move Down toolbar buttons.

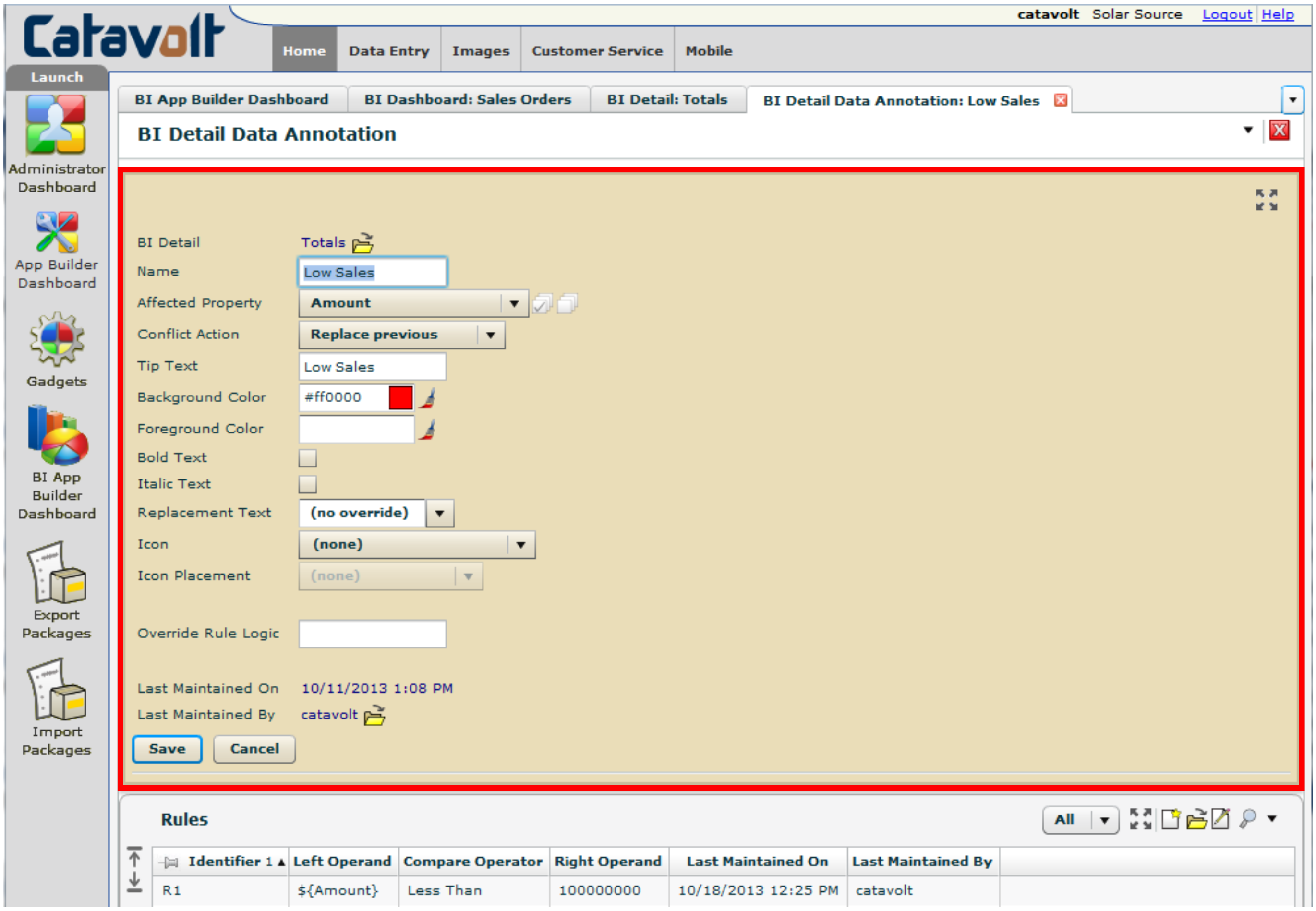


Dashboard	Sales Orders
Name	Totals
Label	Total Sales
Override Range	to

Measurement	Label
Amount	Amount

Name	Sample	Background Color	Foreground Color	Bold Text	Italic Text	Replacement Text	Tip Text	Icon Placement
Low Sales	Sample	#ff0000		No	No		Low Sales	
Overdue	Sample			Yes	No		Overdue	

Figure 69: BI Detail details view with the Data Annotations section highlighted



BI Detail Data Annotation

BI Detail	Totals
Name	Low Sales
Affected Property	Amount
Conflict Action	Replace previous
Tip Text	Low Sales
Background Color	#ff0000 🟥
Foreground Color	
Bold Text	□
Italic Text	□
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)
Override Rule Logic	
Last Maintained On	10/11/2013 1:08 PM
Last Maintained By	catavolt

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand	Last Maintained On	Last Maintained By
R1	${Amount}	Less Than	100000000	10/18/2013 12:25 PM	catavolt

Figure 70: BI Detail Data Annotation create details view

When creating a Data Annotation, you must specify a Name. This value is used to uniquely identify the Data Annotation and is not presented to the end user.

Affected Property specifies which Property(s) will be altered by the Data Annotation or if it applies to the whole section. Xalt defaults this value to (Entire Row), which will alter every Property in the section. You also have the option of choosing one or more Properties from the Detail Property Section.

Conflict Action specifies what should be done if multiple Data Annotations apply to a property. As stated above, Annotations are processed in a specific sequence. Two annotations are in conflict of each other when they both pass all their rules and are annotating the same property or the entire section. When annotations are in conflict, the last one in the list wins. This option allows the annotation to either be merged with previous annotations or replace previous annotations. When an annotation is merged with a previous annotation, it will assume all previous annotations' properties that it does not specify itself. For example, if one annotation sets the background color and a latter one specifies a foreground color and they are merged, the resulting annotation will have both a background and a foreground color set.

Tip Text specifies text that will be displayed when the end user hovers over a property currently affected by a Data Annotation. In this way, the end user can get a text description of the Data Annotation that is currently being applied to the row or property without having to remember what the different colors, fonts, etc. mean when looking at a Query.

Background Color specifies the color you wish to change the background of the affected cell or the entire section if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Foreground Color specifies the color you wish to change the foreground text of the affected cell or the entire section if the Data Annotation is applied. Press the Select Color button to choose an appropriate color, or leave blank to keep the color unchanged.

Bold Text specifies whether you want to change the font to be bold. Select the checkbox to apply a bold font to the affected cell or the entire section if the Data Annotation is applied.

Italic Text specifies whether you want to change the font to be italicized. Select the checkbox to apply an italicized font to the affected cell or the entire section if the Data Annotation is applied.

Replacement Text specifies whether you want to replace the value of the affected cell with an alternate text string. This is commonly used to improve readability of the Detail (for example, to replace 0 values with blanks). Note that Replacement Text is not available when the Affected Property is (Entire Row).

Icon specifies whether you want to append or replace the value of the affected cell with an image. The image you choose can be shown instead of or in addition to the value of the affected cell. Note that Icon is not available when the Affected Property is (Entire Row).

Icon Placement specifies where to place the image in relation to the value of the affected cell. The available values are:

- Replace Text – Replace the value of the affected cell with the image.
- Left of Text – Put the image to the left of the value of the affected cell.
- Right of Text – Put the image to the right of the value of the affected cell.
- Background – Put the image in the background under the value of the affected cell. This value is not used on mobile clients.
- Background (Fill) – Put the image in the background under the value of the affected cell and stretch the image to fill the column horizontally. This is typically used for annotations that show progress bars or something similar. This value is not used on mobile clients

Override Rule Logic specifies more advanced logic to evaluate the Data Annotation Rules in your Rules section. By default, all rules will be evaluated and ANDed together. This means all rules must pass in order for the Data Annotation to be applied. You can use the Override Rule Logic field to apply AND, OR, NOT, and parentheses to allow for more complex logic. You must use the Rule Identifier in the logic expression. For example, if you have 4 Rules defined (R1, R2, R3, and R4), you can define an Override Rule Logic field of

(R1 AND R2) OR (R3 AND R4)

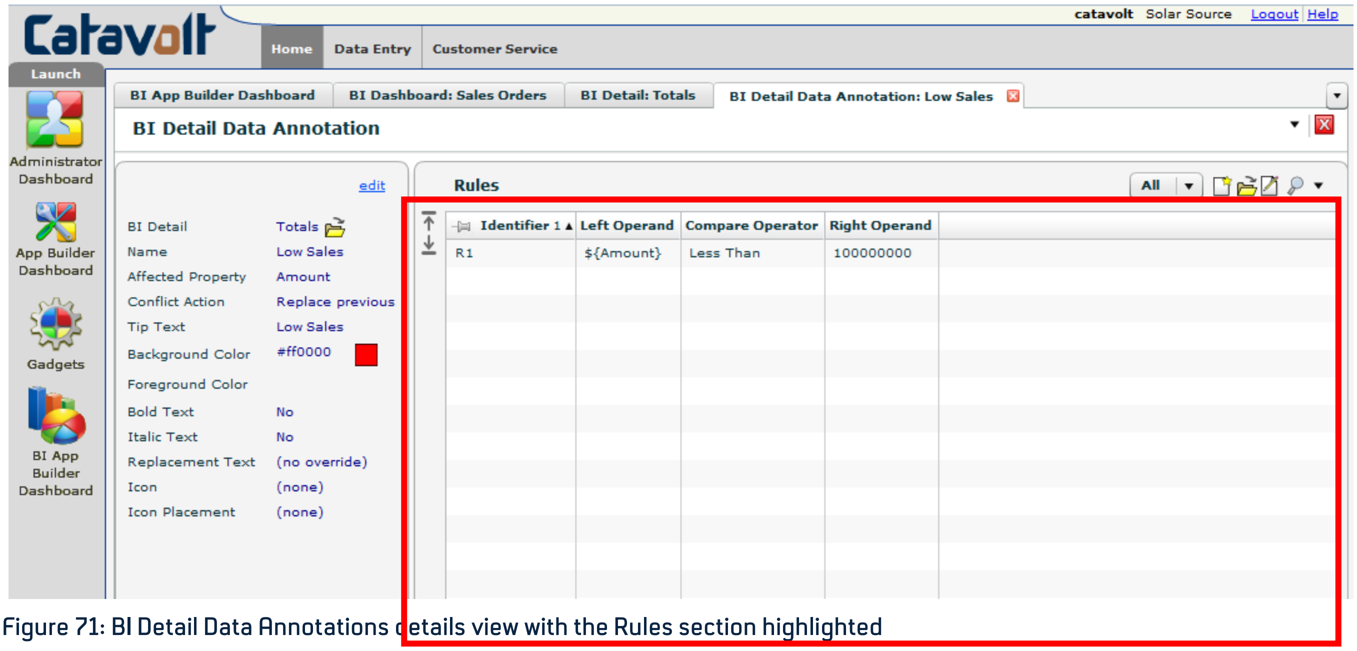
NOT R1 OR (R1 AND NOT R2) OR (R3 AND R4)

R1 OR ((R2 OR R3) AND NOT R4)

and so on. Note that if you do not use parentheses, ANDs will be applied before ORs when evaluating rules.

Data Annotation Rules

The Rules list on a Data Annotation can be used to control when the Data Annotation should be applied to properties. A Data Annotation can have zero or more rules. All rules must pass for the Data Annotation to be applied to that particular row of data. If a Data Annotation has no rules, then it is considered to pass automatically.



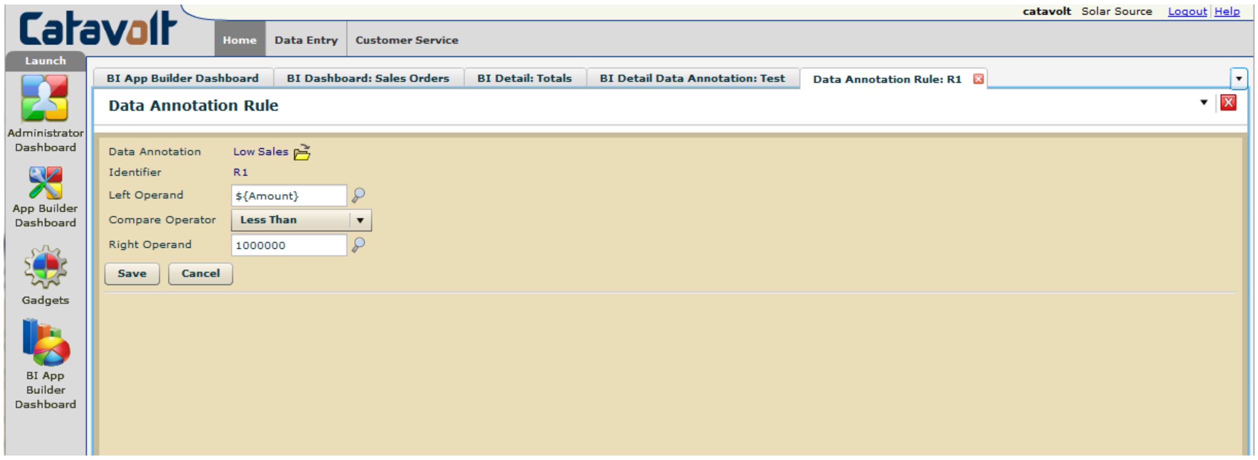
BI Detail Data Annotation

BI Detail	Totals
Name	Low Sales
Affected Property	Amount
Conflict Action	Replace previous
Tip Text	Low Sales
Background Color	#ff0000 🟥
Foreground Color	
Bold Text	No
Italic Text	No
Replacement Text	(no override)
Icon	(none)
Icon Placement	(none)

Rules

Identifier 1 ▲	Left Operand	Compare Operator	Right Operand
R1	${Amount}	Less Than	100000000

Figure 71: BI Detail Data Annotations details view with the Rules section highlighted



Data Annotation Rule

Data Annotation	Low Sales
Identifier	R1
Left Operand	${Amount}
Compare Operator	Less Than
Right Operand	1000000

Figure 72: BI Details Data Annotation Rule create details view

When creating a Data Annotation Rule, the Xalt will automatically create a unique Identifier for the Rule.

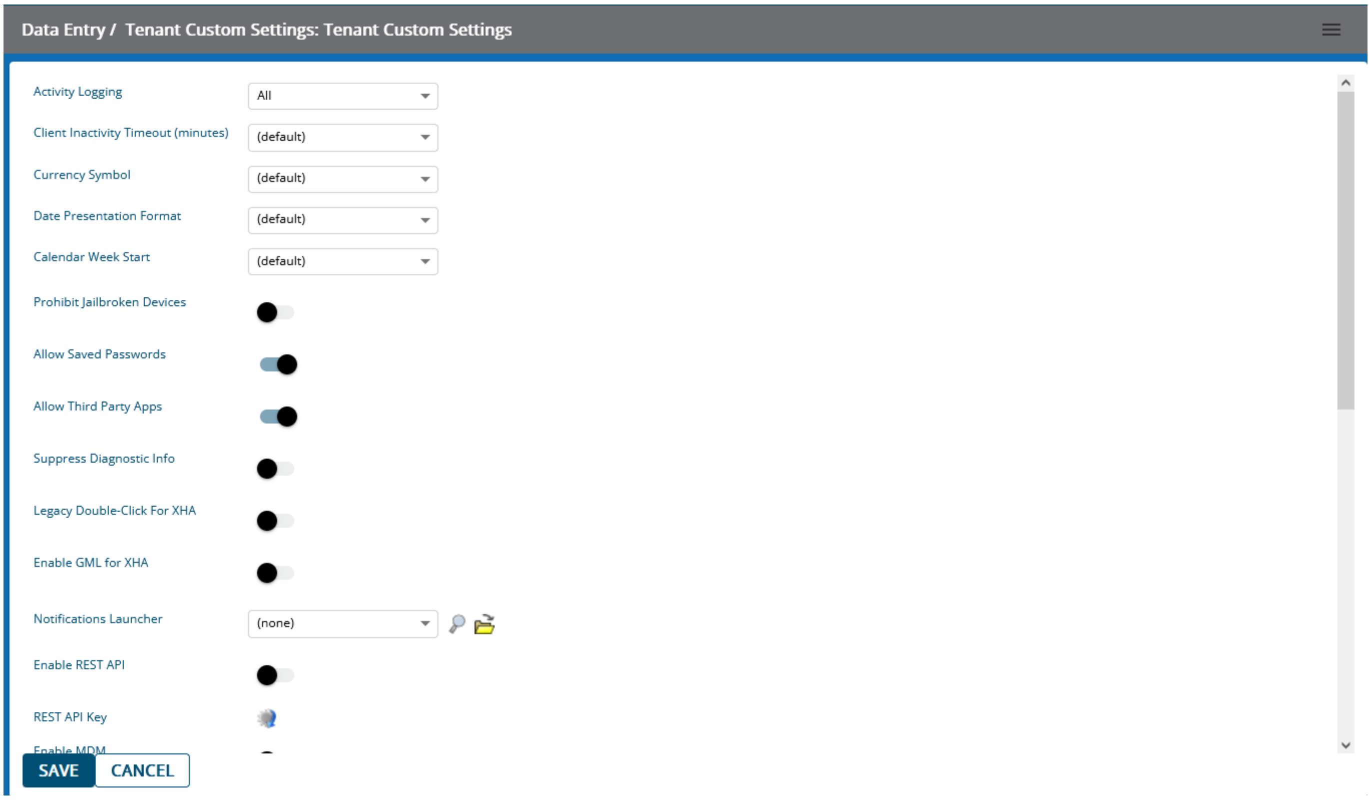
Left Operand and Right Operand specify the two values that will be compared with each other. You have 2 choices for operands: Constant Values (such as A or 1.00) or Substitution Values. You can enter a Substitution Value manually or press the Find button to have Xalt build a Substitution Value based on the Available Properties for the Data Object. Appendix A: Specifying Messages and Substitution Values for more information about how to create Substitution Messages for Left Operand and Right Operand. A special Substitution Value of ${AFFECTED_PROPERTY} is available for annotations where you select multiple Affected Properties. When running the annotation, the rule will be checked independently for each affected property. This allows you to create a single Annotation and a single rule that can apply to multiple individual properties. For example, if you have 5 numeric properties in a Detail Property Section and you want any negative values to show as red, you can create a single annotation to make the text foreground color red, choose all 5 numeric columns as Affected Properties, and create a single Annotation Rule of ${AFFECTED_PROPERTY} Less Than 0.

Compare Operator specifies how the two operands should be compared with each other. Options include Equal, Not Equal, Greater Than, Less Than, Greater or Equal, Less or Equal, Contains, Starts With, Ends With, Is Null, and Is Not Null.

Chapter 12: Custom Settings

Chapter Summary

Xalt | Mobility has added the ability for you to customize certain behavior in your Xalt system. You can access these settings by clicking the Custom Settings launcher on the Data Entry Workbench:



Data Entry / Tenant Custom Settings: Tenant Custom Settings

Activity Logging	All
Client Inactivity Timeout (minutes)	(default)
Currency Symbol	(default)
Date Presentation Format	(default)
Calendar Week Start	(default)
Prohibit Jailbroken Devices	○
Allow Saved Passwords	●
Allow Third Party Apps	●
Suppress Diagnostic Info	○
Legacy Double-Click For XHA	○
Enable GML for XHA	○
Notifications Launcher	(none)
Enable REST API	○
REST API Key	
Enable MDM	

Figure 1: Tenant Custom Settings window

Activity Logging specifies the level of logging that will occur. The available options are:

- **All** – All normal activity will be logged. This is the default setting.
- **Restricted** – Normal activity will be logged, but property before/after values will be omitted.
- **None** – No activity will be logged.

Client Inactivity Timeout specifies the number of minutes a client can remain idle before the user is automatically logged out of Xalt. You can set this to anywhere between 5 and 240 minutes. If you choose (default), Xalt will use its internal default value which is currently set to 60 minutes.

Currency Symbol specifies the symbol that should be used when displaying Money values. You can specify one or more characters to be used for the currency symbol. If you choose (default), Xalt will use the currency symbol specified by the user's locale setting.

Date Presentation Format specifies special formatting options used to display Date values in read mode.

- (default) – Let the client decide which format to use (this is the default).
- Short – Force the client to use a Short Date format (actual format is dependent on client's locale settings)
- Long – Force the client to use a Long Date format (actual format is dependent on client's locale settings)

Calendar Week Start specifies which day should be shown first in Week View and Month View Calendars. You can choose any day of the week. If the value is set to (default), the client will decide which day to use first.

Prohibit Jailbroken Devices specifies whether Xalt | Mobility is allowed to run on jailbroken or rooted mobile devices. If the box is checked, Xalt | Mobility will detect if the mobile device has been jailbroken and, if so, Xalt will not run. The default value for this option is to allow Xalt | Mobility to run on jailbroken mobile devices.

Allow Saved Passwords specifies whether Xalt | Mobility allows the user to save their password credentials on their mobile device for a temporary period of time to not force them to enter their password on every login. If the box is unchecked, the Save Password option will not be available for users. The default value for this option is to allow saved passwords.

Allow Third Party Apps specifies whether Xalt | Mobility allows third party mobile apps to open documents and attachments. If the box is checked, the user will get a popup when downloading a document or attachment to their mobile device. This popup allows the user to send the document to a third party app that can process the document (for example, downloading an AutoCAD drawing would allow the user to open the file directly in the AutoCAD 360 app). If the box is unchecked, downloaded documents or attachments will only be opened inside the Xalt | Mobility app. This also means that document types that Xalt | Mobility cannot handle internally cannot be displayed. In addition, all Export menu options will be removed from Data Objects that you create in Xalt. The default value for this option is to allow third party apps to open documents.

Suppress Diagnostic Info will remove stack traces from error messages. When this option is selected, stack traces will be replaced with a token (for example, ST:357) that Hexagon Support can use to retrieve the original stack trace. Note that as stack traces can be useful to you during application development, running the Enter Development Mode action will re-enable stack traces for the duration of your development session.

Legacy Double-Click for XHA specifies whether the XHA web client should accept a single- or double-click gesture in lists to perform the default action. If this option is unselected (off), the XHA client will use a single-click. If this option is selected (on), the XHA client will use a double-click (same as the legacy Flex rich client).

Enable GML for XHA specifies whether the XHA web client should use GML when rendering screens. If this option is unselected (off), the XHA client will ignore GML when rendering screens. If this option is selected (on), the XHA client will use existing GML when rendering screens.

Notifications Launcher specifies the Xalt Data Object to launch when the user chooses to display their Notifications. Your Hexagon Professional Services representative has more information about configuring and using Hexagon Notifications.

Enable REST API specifies whether to turn on a REST API for your tenant to expose your Data Objects. More information about this option can be found in the **Xalt | Mobility REST API Guide**.

REST API Key specifies API Key for your tenant to use when accessing the Xalt | Mobility REST API. More information about this option can be found in the **Xalt | Mobility REST API Guide**.

Enable MDM specifies whether to turn on MDM (Mobile Device Management) for your tenant. When you turn this on, only mobile devices that you have specifically deployed with the correct MDM Key will be allowed to connect to Xalt | Mobility. **PLEASE NOTE:** You should only turn this on if you are deploying the MDM version of the Hexagon mobile application. The regular Xalt | Mobility application will no longer work once you enable this option.

MDM Key specifies the MDM Key for your tenant to use when accessing Xalt | Mobility via a mobile device. This MDM Key will be specified when configuring your application for deployment via MDM. You can generate a different key by pressing Cancel and then pressing the “Generate New MDM Key” button.

The **User Password Requirements** section allows you to specify a variety of password complexity rules for end users who use the default (Xalt) Authentication Service. When new passwords are created for your tenant, the following complexity rules are available:

Minimum Length specifies the minimum length for new passwords. This value can be between 6 and 20. The default value is 8.

Password History specifies the number of previous passwords that the new password cannot be same as. This value can be between 0 and 10. 0 indicates the new password cannot be the same as the current password. A value between 1 and 10 indicates that this many previous passwords will also be checked. The default value is 0. Note that this rule is only checked when a User changes their own password. An administrator is allowed to reset a user's password to a previously used value if desired.

Maximum Repeated Characters specifies the maximum number of times the same character can be repeated in a new password. For example, a password of **paaaasswooord** has the character 'a' repeated 4 times, 's' 2 times, and 'o' 3 times. This value can be between 0 and 20. 0 indicates the same character can be repeated without limit. The default value is 2.

Uppercase Character Required specifies whether the new password must contain at least 1 Uppercase character. The default value is Yes.

Lowercase Character Required specifies whether the new password must contain at least 1 Lowercase character. The default value is Yes.

Digit Required specifies whether the new password must contain at least 1 numeric character. The default value is Yes.

Special Character Required specifies whether the new password must contain at least 1 Special character. Any character that is not a letter or digit is considered a Special Character. The default value is No.

Cannot Contain E-mail Address specifies that the new password cannot contain any part of the User Profile's email address, including reversed values and alternate case. Email parts are broken up using the period (.) and @ characters. The final suffix of the email address (com, org, etc) will be ignored. Any part that is only 2 characters or less will also be ignored. If a User Profile does not have an Email Address specified, this check will be skipped.

For example, if a User Profile's email address is **jane@doe.com**, the following values cannot be included in the new password: **jane, enaj, JaNe, ENAJ, doe, EOD**, etc. If a User Profile's email address is wile.e.coyote@acme.com the following values cannot included in the new password: **wile, eLiW, coyote, acme, EMCA**, etc. The default value is No (do not check).

Cannot Contain Tenant / User ID specifies that the new password cannot contain your Tenant ID or the User Profile's User ID, including reversed values and alternate case.

For example, if your tenant ID is **acme** and the User Profile's User ID is **jeffo**, the following values cannot included in the new password: **acme, emca, EMCA, jeffo, OFfEJ**, etc. The default value is No (do not check).

Chapter 13: Image/Asset Management

Chapter Summary

Xalt | Mobility uses images and external files in a number of places, including Launchers, GML, Forms, etc. To help better manage the external images that are being used, Xalt has added an Images workbench that contains launchers to view and manage these assets.

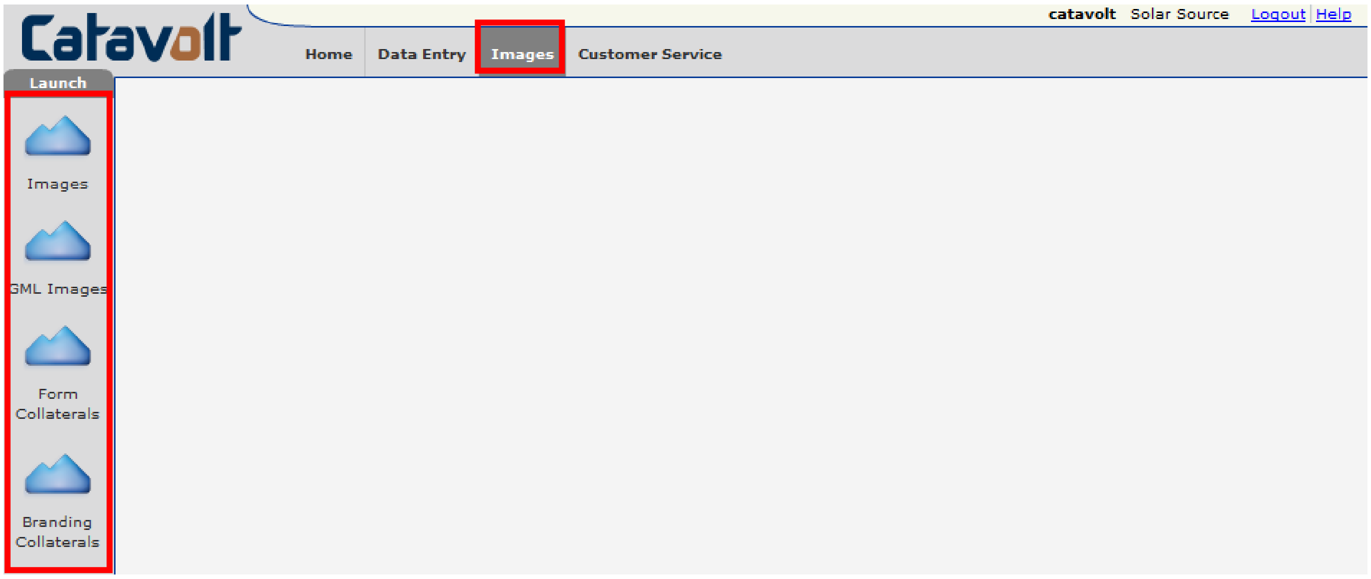


Figure 1: Images workbench

Available Images

Choosing the Images launcher will display a list of Available Images. These images are available to be used for icons and data annotations.

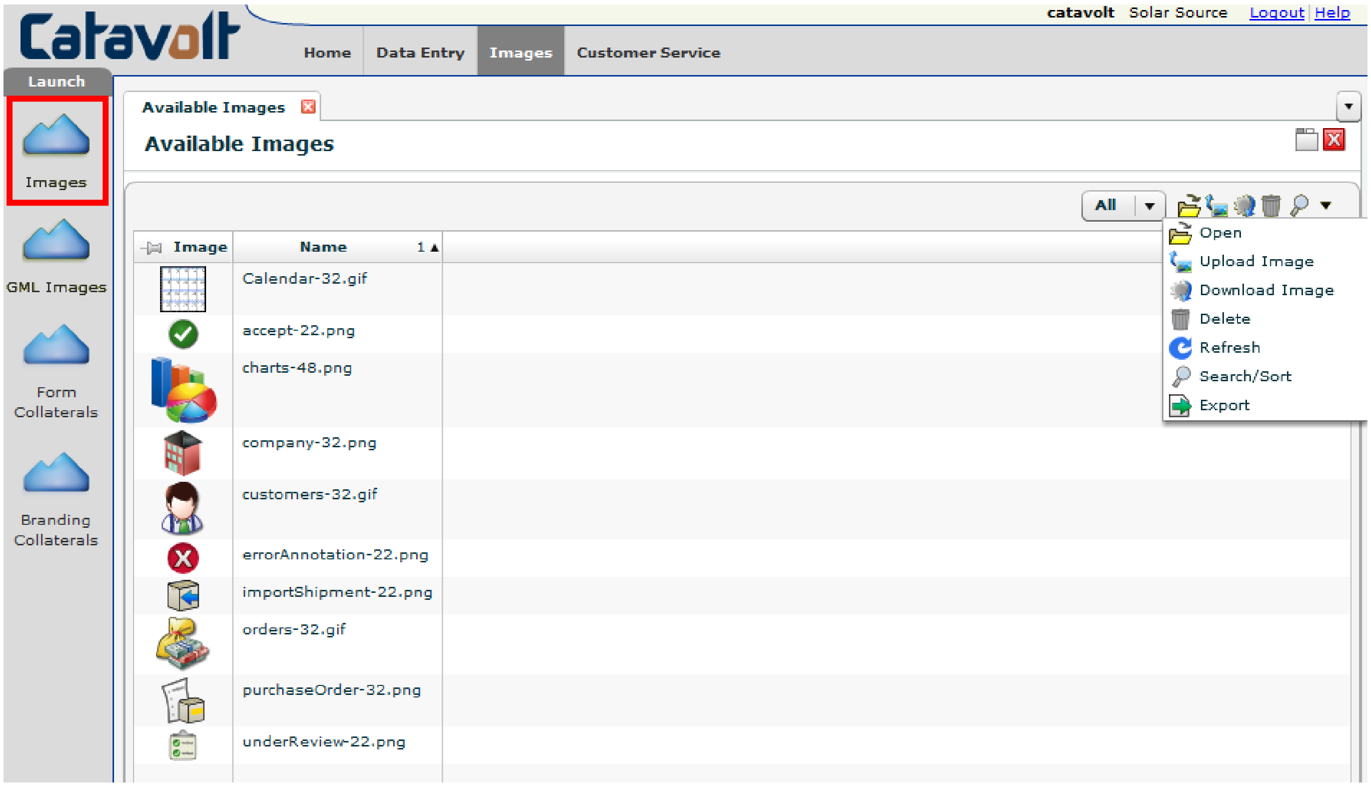


Image	Name 1 ▲
	Calendar-32.gif
	accept-22.png
	charts-48.png
	company-32.png
	customers-32.gif
	errorAnnotation-22.png
	importShipment-22.png
	orders-32.gif
	purchaseOrder-32.png
	underReview-22.png

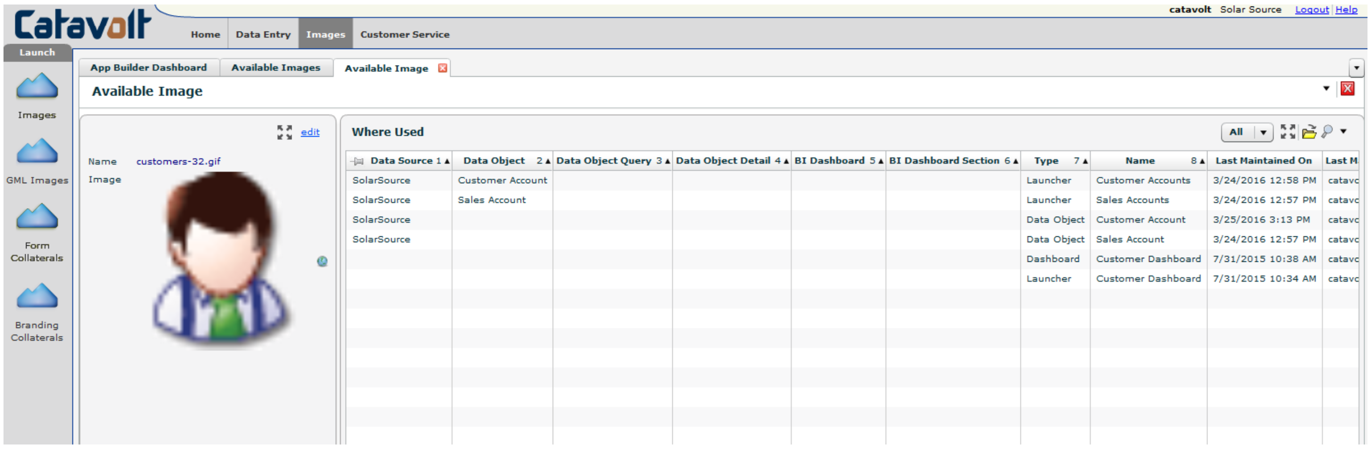
Figure 2: Available Images list

You can perform Download Image to retrieve the selected image from the server. If you select multiple images, all selected images will be packaged together in a ZIP file.

You can perform Upload Image to add a new image to the list of available images. If you select a ZIP file, it will be unzipped and all files inside will be uploaded.

You can perform Delete to remove one or more images from the list of available images. When you delete an image, Xalt will find and blank out all references to specified image.

If you open a record, you will see details that contains a Where Used list of objects that are currently using the image:



Available Image

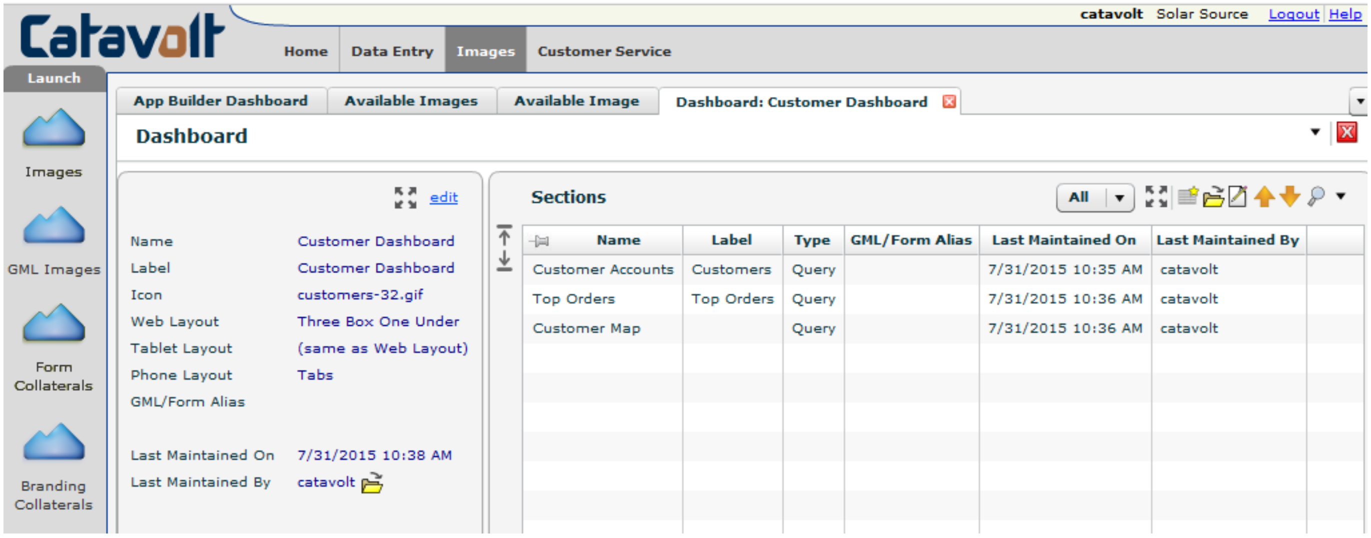
Name customers-32.gif
Image

Where Used

Data Source 1 ▲	Data Object 2 ▲	Data Object Query 3 ▲	Data Object Detail 4 ▲	BI Dashboard 5 ▲	BI Dashboard Section 6 ▲	Type 7 ▲	Name 8 ▲	Last Maintained On	Last M
SolarSource	Customer Account					Launcher	Customer Accounts	3/24/2016 12:58 PM	catavc
SolarSource	Sales Account					Launcher	Sales Accounts	3/24/2016 12:57 PM	catavc
SolarSource						Data Object	Customer Account	3/25/2016 3:13 PM	catavc
SolarSource						Data Object	Sales Account	3/24/2016 12:57 PM	catavc
						Dashboard	Customer Dashboard	7/31/2015 10:38 AM	catavc
						Launcher	Customer Dashboard	7/31/2015 10:34 AM	catavc

Figure 3: Available Images detail

Opening a Where Used record will take you directly to the object using the image:



Dashboard

Name Customer Dashboard
Label Customer Dashboard
Icon customers-32.gif
Web Layout Three Box One Under
Tablet Layout (same as Web Layout)
Phone Layout Tabs
GML/Form Alias

Last Maintained On 7/31/2015 10:38 AM
Last Maintained By catavolt

Sections

Name	Label	Type	GML/Form Alias	Last Maintained On	Last Maintained By
Customer Accounts	Customers	Query		7/31/2015 10:35 AM	catavolt
Top Orders	Top Orders	Query		7/31/2015 10:36 AM	catavolt
Customer Map		Query		7/31/2015 10:36 AM	catavolt

Figure 4: Images Where Used object

Available GML Images

Choosing the GML Images launcher will display a list of Available GML Images. These images are available to be used for GML re-stylizing of mobile screens. Your Hexagon sales representative has more information on how to utilize GML on your mobile device.

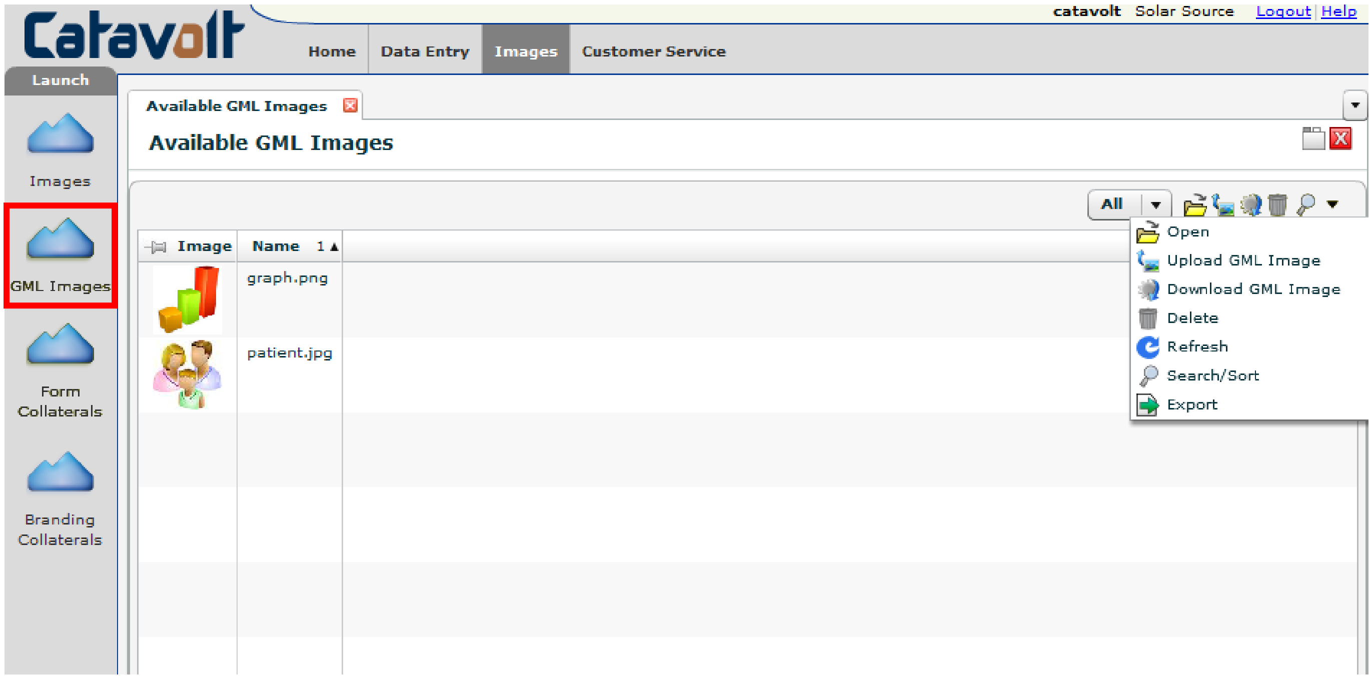


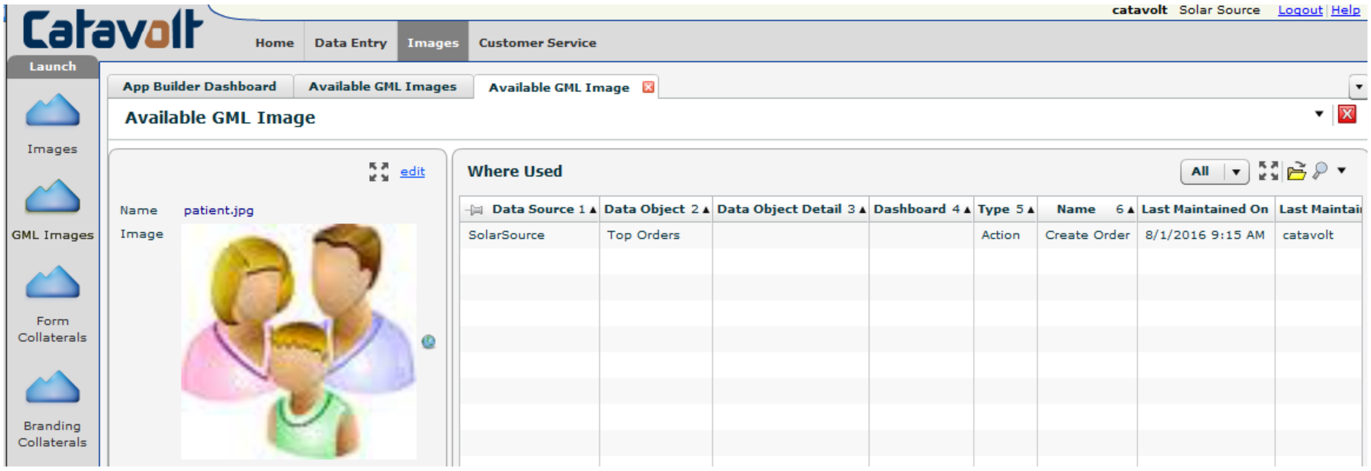
Figure 5: Available GML Images list

You can perform Download GML Image to retrieve the selected GML image from the server. If you select multiple images, all selected images will be packaged together in a ZIP file.

You can perform Upload GML Image to add a new GML image to the list of available images. If you select a ZIP file, it will be unzipped and all files inside will be uploaded.

You can perform Delete to remove one or more images from the list of available GML images. When you delete an image, Xalt will check all references to specified image. If any references exist, you will be notified and asked to remove the references. If no references exist, the image will be removed.

If you open a record, you will see details that contains a Where Used list of objects that are currently using the image:



Available GML Image

Name patient.jpg

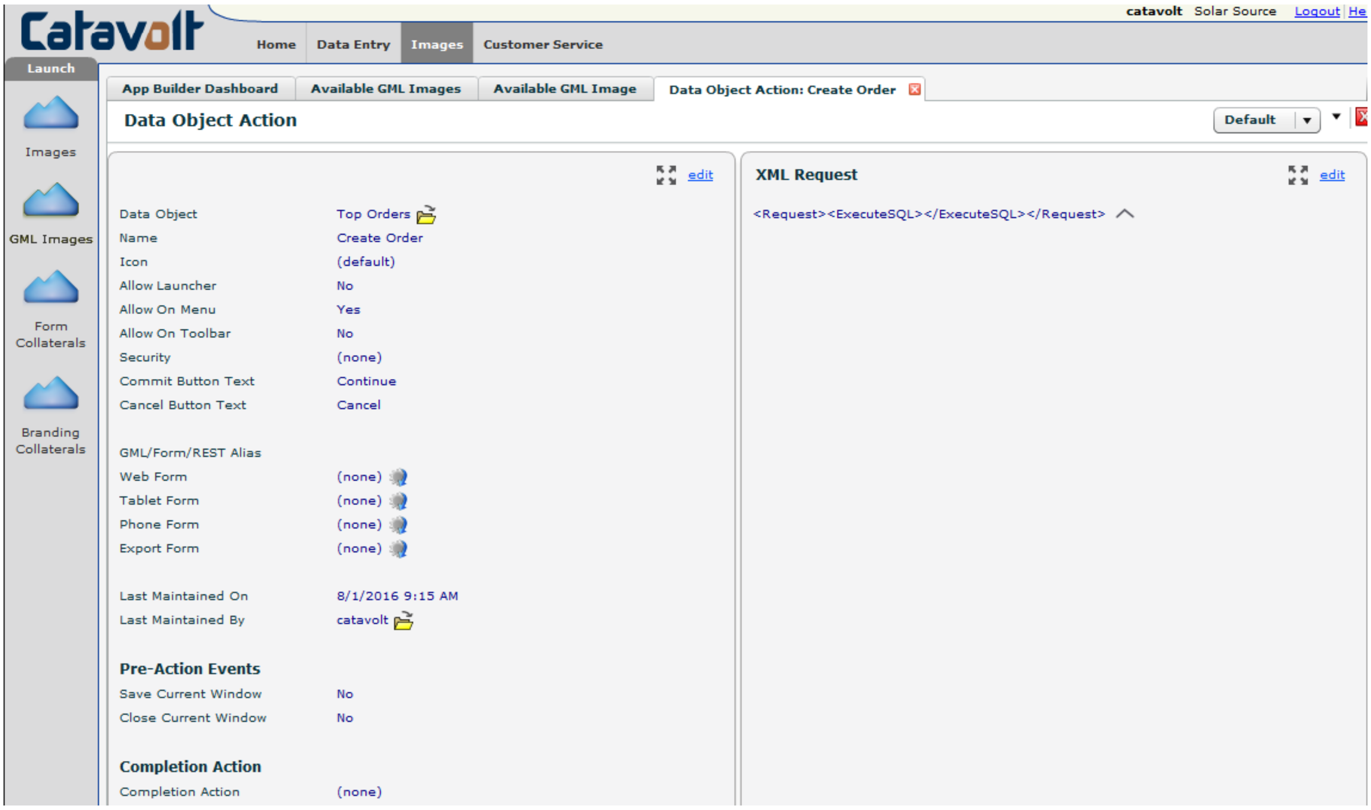
Image

Where Used

Data Source 1	Data Object 2	Data Object Detail 3	Dashboard 4	Type 5	Name 6	Last Maintained On	Last Maintai
SolarSource	Top Orders			Action	Create Order	8/1/2016 9:15 AM	catavolt

Figure 6: Available GML Images detail

Opening a Where Used record will take you directly to the object using the image:



Data Object Action

Data Object	Top Orders
Name	Create Order
Icon	(default)
Allow Launcher	No
Allow On Menu	Yes
Allow On Toolbar	No
Security	(none)
Commit Button Text	Continue
Cancel Button Text	Cancel
GML/Form/REST Alias	
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	8/1/2016 9:15 AM
Last Maintained By	catavolt

Pre-Action Events

Save Current Window	No
Close Current Window	No

Completion Action

Completion Action	(none)

XML Request

```
<Request><ExecuteSQL></ExecuteSQL></Request>
```

Figure 7: GML Images Where Used object

Available Form Collaterals

Choosing the Form Collaterals launcher will display a list of Available Form Collaterals (Xalt custom forms and images on these forms). These collaterals are available to be used for Xalt Custom forms. Your Hexagon sales representative has more information on how to create and deploy Custom Forms.

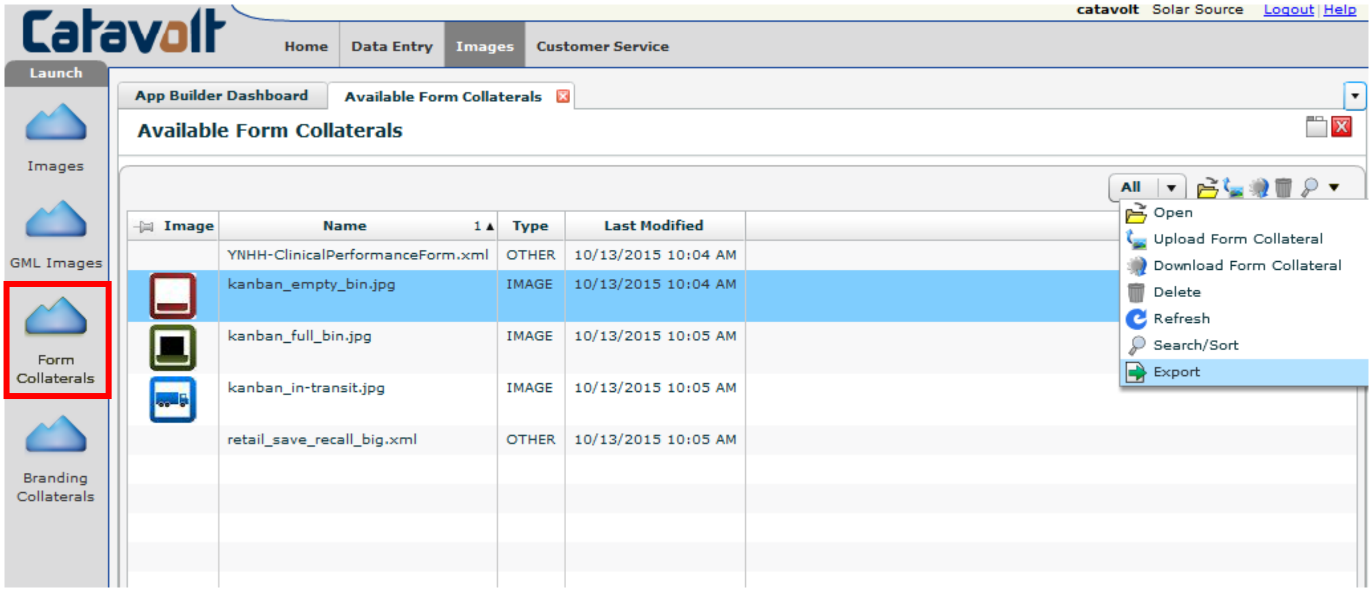


Image	Name	Type	Last Modified
	YNHH-ClinicalPerformanceForm.xml	OTHER	10/13/2015 10:04 AM
	kanban_empty_bin.jpg	IMAGE	10/13/2015 10:04 AM
	kanban_full_bin.jpg	IMAGE	10/13/2015 10:05 AM
	kanban_in-transit.jpg	IMAGE	10/13/2015 10:05 AM
	retail_save_recall_big.xml	OTHER	10/13/2015 10:05 AM

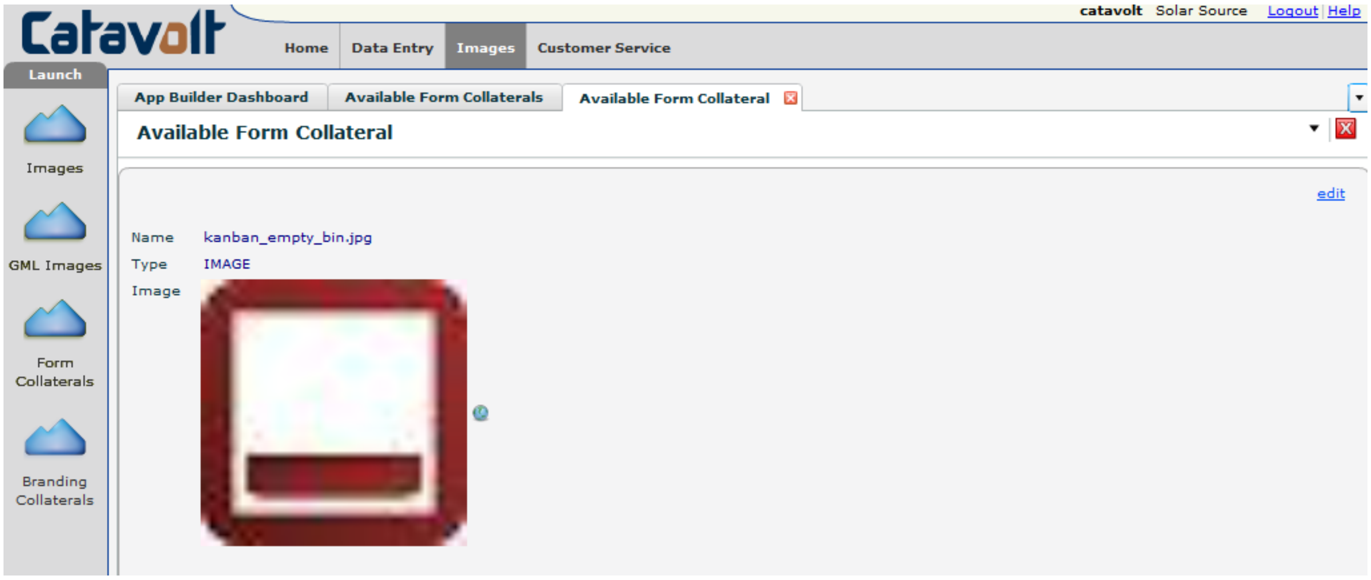
Figure 6: Available Form Collaterals list

You can perform Download Form Collateral to retrieve the selected Xalt custom form or image from the server. If you select multiple files, all selected files will be packaged together in a ZIP file.

You can perform Upload Form Collateral to add a new Xalt custom form or image to the list of available collaterals. If you select a ZIP file, it will be unzipped and all files inside will be uploaded.

You can perform Delete to remove a collateral from the list of available collaterals. Images will be automatically deleted. When you delete a form, Xalt will find and blank out all references to specified form.

If you open an image file, you will see the following details:



Available Form Collateral

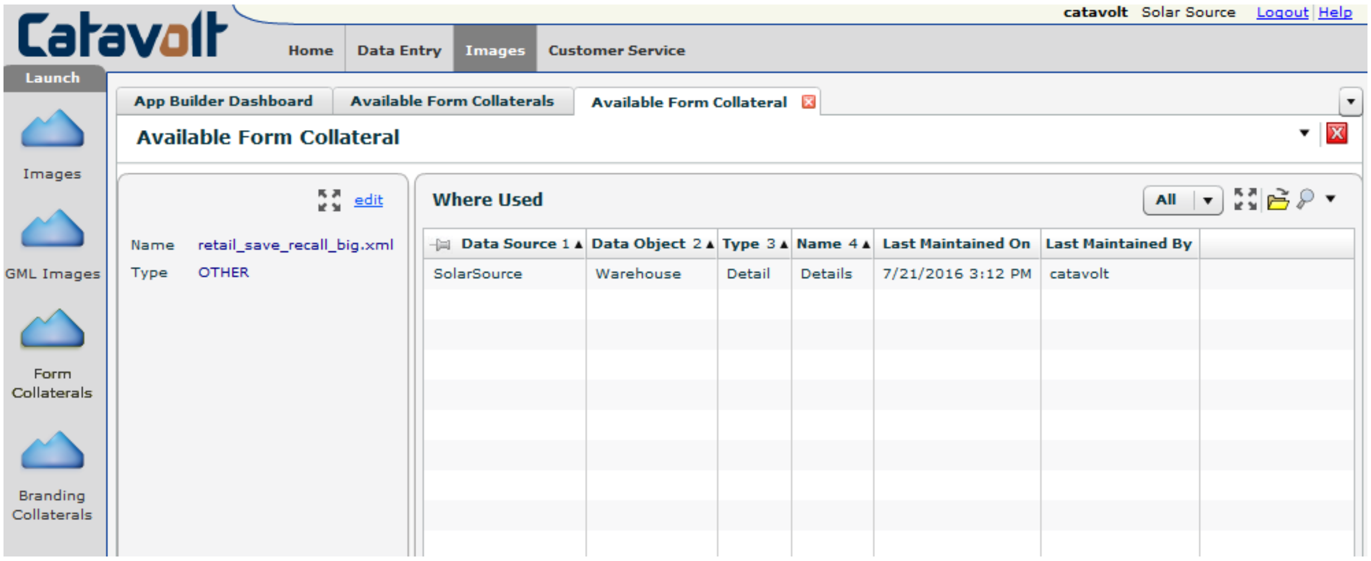
Name kanban_empty_bin.jpg

Type IMAGE

Image

Figure 7: Available Form Collateral detail (image)

If you open an xml file, you will see details that contains a Where Used list of objects (Details and Actions) that are currently using the form:



Available Form Collateral

Name retail_save_recall_big.xml

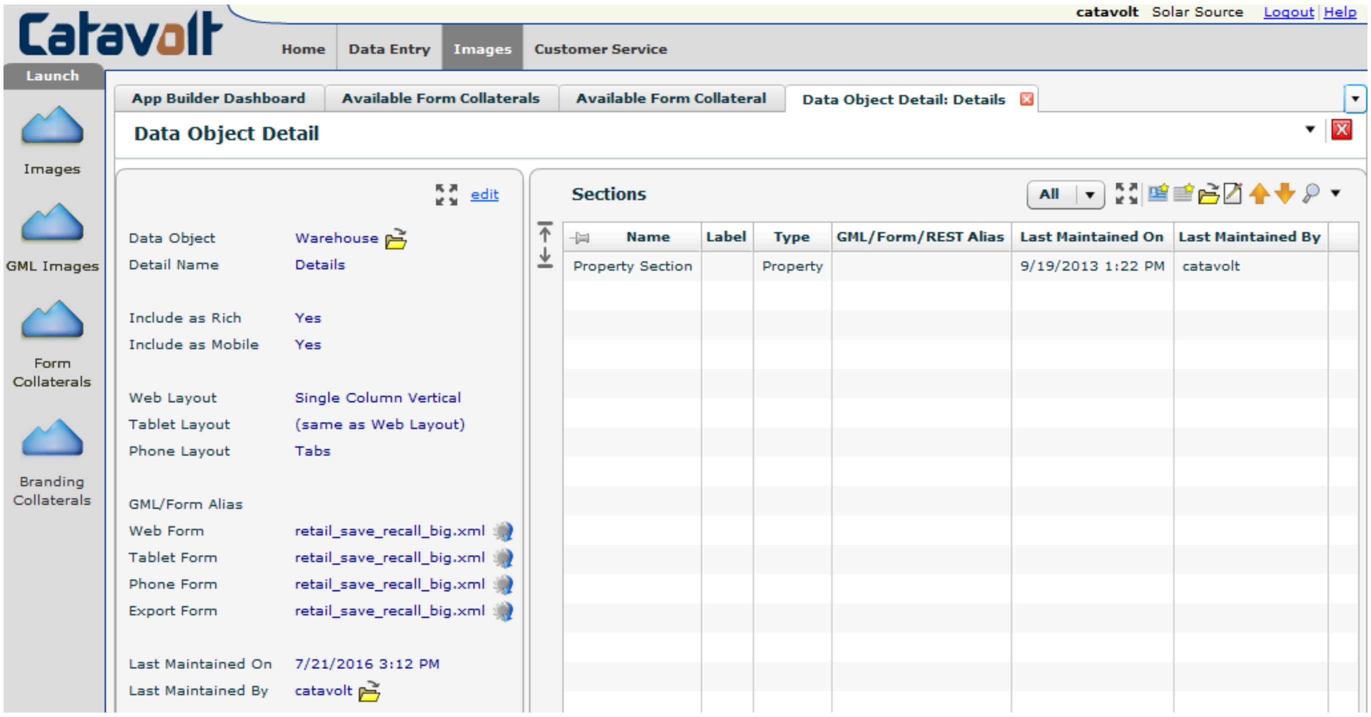
Type OTHER

Where Used

Data Source 1 ▲	Data Object 2 ▲	Type 3 ▲	Name 4 ▲	Last Maintained On	Last Maintained By
SolarSource	Warehouse	Detail	Details	7/21/2016 3:12 PM	catavolt

Figure 8: Available Form Collateral detail (form)

Opening a Where Used record will take you directly to the object using the form:



Data Object Detail

Data Object	Warehouse
Detail Name	Details
Include as Rich	Yes
Include as Mobile	Yes
Web Layout	Single Column Vertical
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Web Form	retail_save_recall_big.xml
Tablet Form	retail_save_recall_big.xml
Phone Form	retail_save_recall_big.xml
Export Form	retail_save_recall_big.xml
Last Maintained On	7/21/2016 3:12 PM
Last Maintained By	catavolt

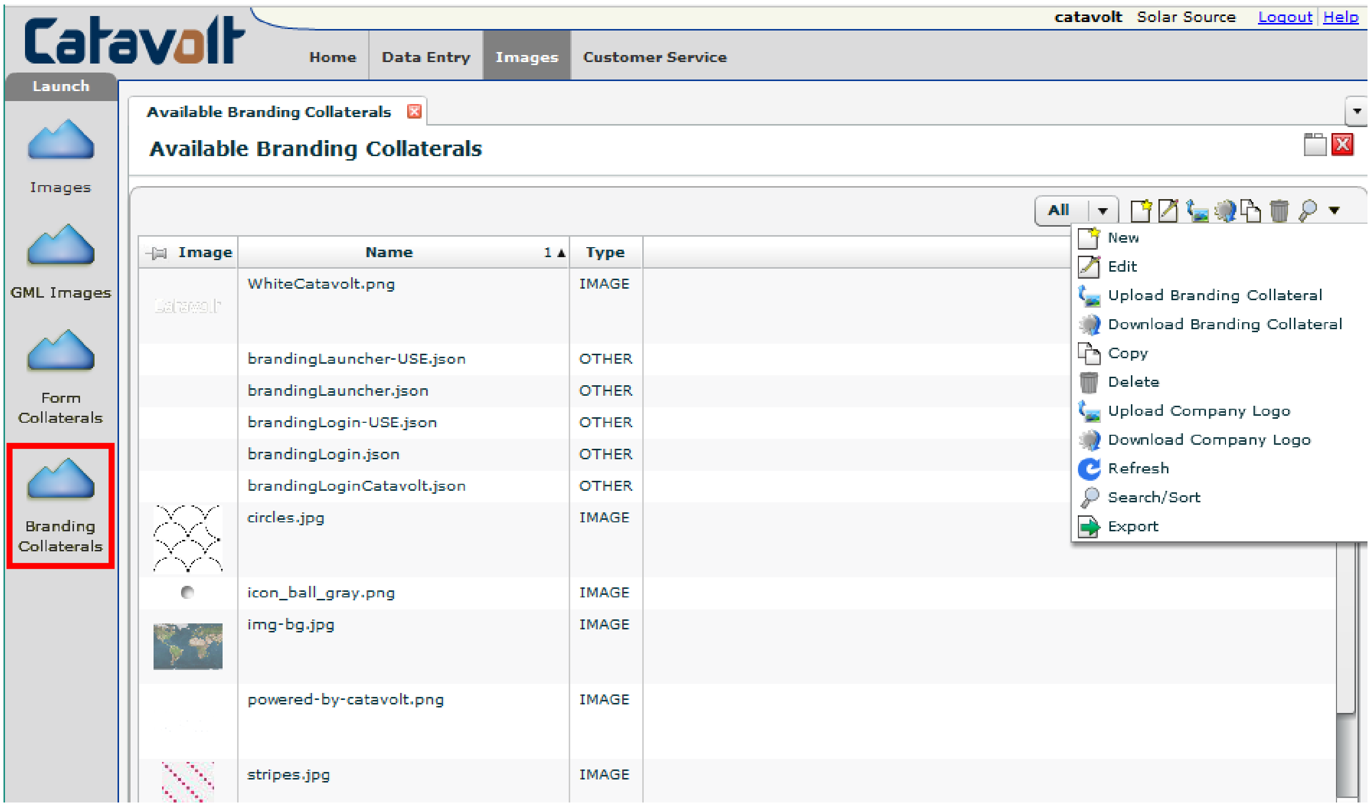
Sections

Name	Label	Type	GML/Form/REST Alias	Last Maintained On	Last Maintained By
Property Section		Property		9/19/2013 1:22 PM	catavolt

Figure 9: Available Form Collateral Where Used

Available Branding Collaterals

Choosing the Branding Collaterals launcher will display a list of Available Branding Collaterals. These collaterals are available to be used for Login and Workbench styling for mobile devices. You would typically use this to brand Xalt with your own company logo/color scheme.



Available Branding Collaterals

Image	Name 1 ▲	Type
	WhiteCatavolt.png	IMAGE
	brandingLauncher-USE.json	OTHER
	brandingLauncher.json	OTHER
	brandingLogin-USE.json	OTHER
	brandingLogin.json	OTHER
	brandingLoginCatavolt.json	OTHER
	circles.jpg	IMAGE
	icon_ball_gray.png	IMAGE
	img-bg.jpg	IMAGE
	powered-by-catavolt.png	IMAGE
	stripes.jpg	IMAGE

Figure 10: Available Branding Collaterals list

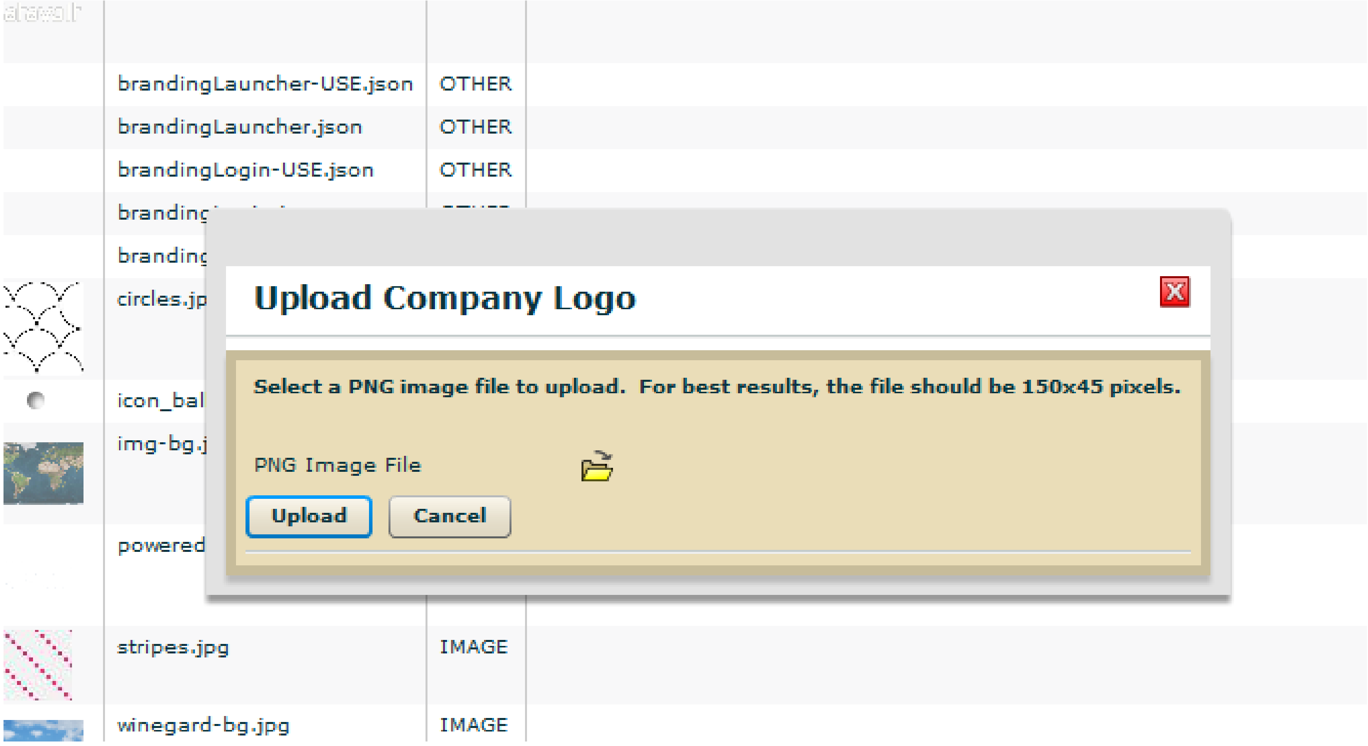
You can perform Download Branding Collateral to retrieve the selected branding file or image from the server. If you select multiple files, all selected files will be packaged together in a ZIP file.

You can perform Upload Branding Collateral to add a new branding file or image to the list of available collaterals. If you select a ZIP file, it will be unzipped and all files inside will be uploaded.

You can perform Delete to remove a collateral from the list of available collaterals.

You can perform Download Company Logo to retrieve the Company Logo image from the server.

You can perform Upload Company Logo to set the Company Logo image. It is recommended that you upload a .PNG file with dimensions of 150x45 for best results.



	brandingLauncher-USE.json	OTHER
	brandingLauncher.json	OTHER
	brandingLogin-USE.json	OTHER
	branding	
	branding	
	circles.jp	
	icon_bal	
	img-bg.j	
	powered	
	stripes.jpg	IMAGE
	winegard-bg.jpg	IMAGE

Upload Company Logo

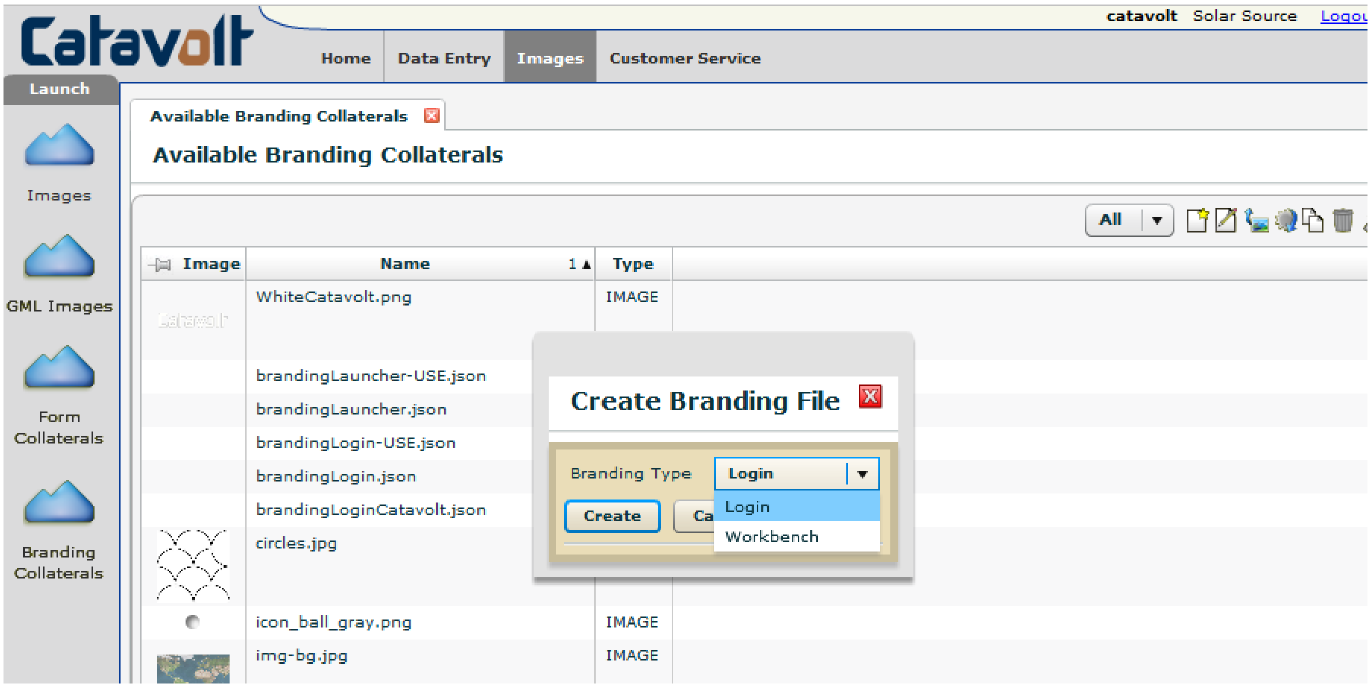
Select a PNG image file to upload. For best results, the file should be 150x45 pixels.

PNG Image File

Figure 11: Upload Company Logo dialog

Creating a Branding File

Selecting New will allow you to create a new Branding file to style your mobile device.



Available Branding Collaterals

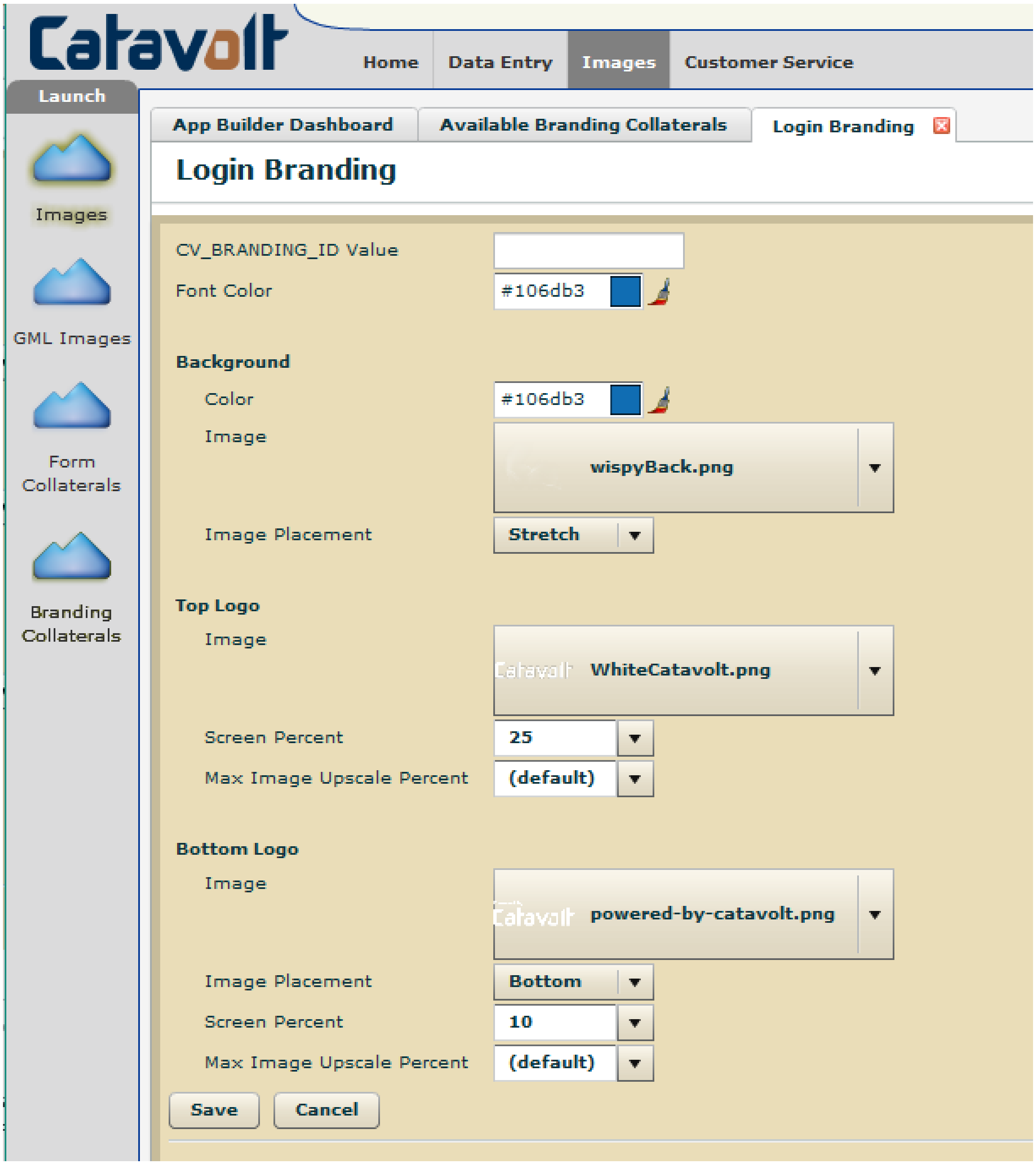
Image	Name	Type
	WhiteCatavolt.png	IMAGE
	brandingLauncher-USE.json	
	brandingLauncher.json	
	brandingLogin-USE.json	
	brandingLogin.json	
	brandingLoginCatavolt.json	
	circles.jpg	
	icon_ball_gray.png	IMAGE
	img-bg.jpg	IMAGE

Branding Type: Login (options: Login, Workbench)

Figure 12: Create Branding File dialog

Branding Type specifies whether the new file should be for the login or the workbench page. Each option will present a different dialog for editing.

If you select Login, you will be presented with the following dialog:



Login Branding

CV_BRANDING_ID Value	
Font Color	#106db3
Background	
Color	#106db3
Image	wispyBack.png
Image Placement	Stretch
Top Logo	
Image	WhiteCatavolt.png
Screen Percent	25
Max Image Upscale Percent	(default)
Bottom Logo	
Image	powered-by-catavolt.png
Image Placement	Bottom
Screen Percent	10
Max Image Upscale Percent	(default)

Figure 13: Create Login Branding File dialog

CV_BRANDING_ID Value allows you to specify an alternate branding ID. If you leave this value blank, a brandingLogin.json file will be created that will specify the Login page branding for all users. If you want to provide alternate branding for specific users or classes of user, specify a value for this property. A brandingLogin-<value>.json file will be created to specify the Login page branding (see note below).

Font Color specifies the font color for the labels of the login prompt. Leaving this value blank will use the default color.

Background Color specifies the background color of the login prompt. Leaving this value blank will use the default color.

Background Image specifies an image to be used as the background of the login prompt. Leaving this value blank will use no image.

Background Image Placement specifies where to place the background image of the login prompt. Leaving this value blank will use the default placement. Allowable values are:

- Stretch – Stretch the image to fill the screen.
- Tile – Tile multiple copies of the image as needed to fill the screen.
- Fit – Adjust the image to fit the screen.

Top Logo Image specifies an image to be used at the top of the login prompt. Leaving this value blank will use no image.

Top Logo Screen Percent specifies the vertical percent (1-100) of the screen that the top logo should occupy. Leaving this value blank will use the default percentage. This value is typically set from 20 to 40 percent.

Top Logo Max Upscale Percent specifies the max percentage (101-9999) that the image can be upscaled in order to fit on the space required. Leaving this value blank will keep the image at 100% (no upscaling).

Bottom Logo Image specifies an image to be used at the bottom of the login prompt. Leaving this value blank will use no image.

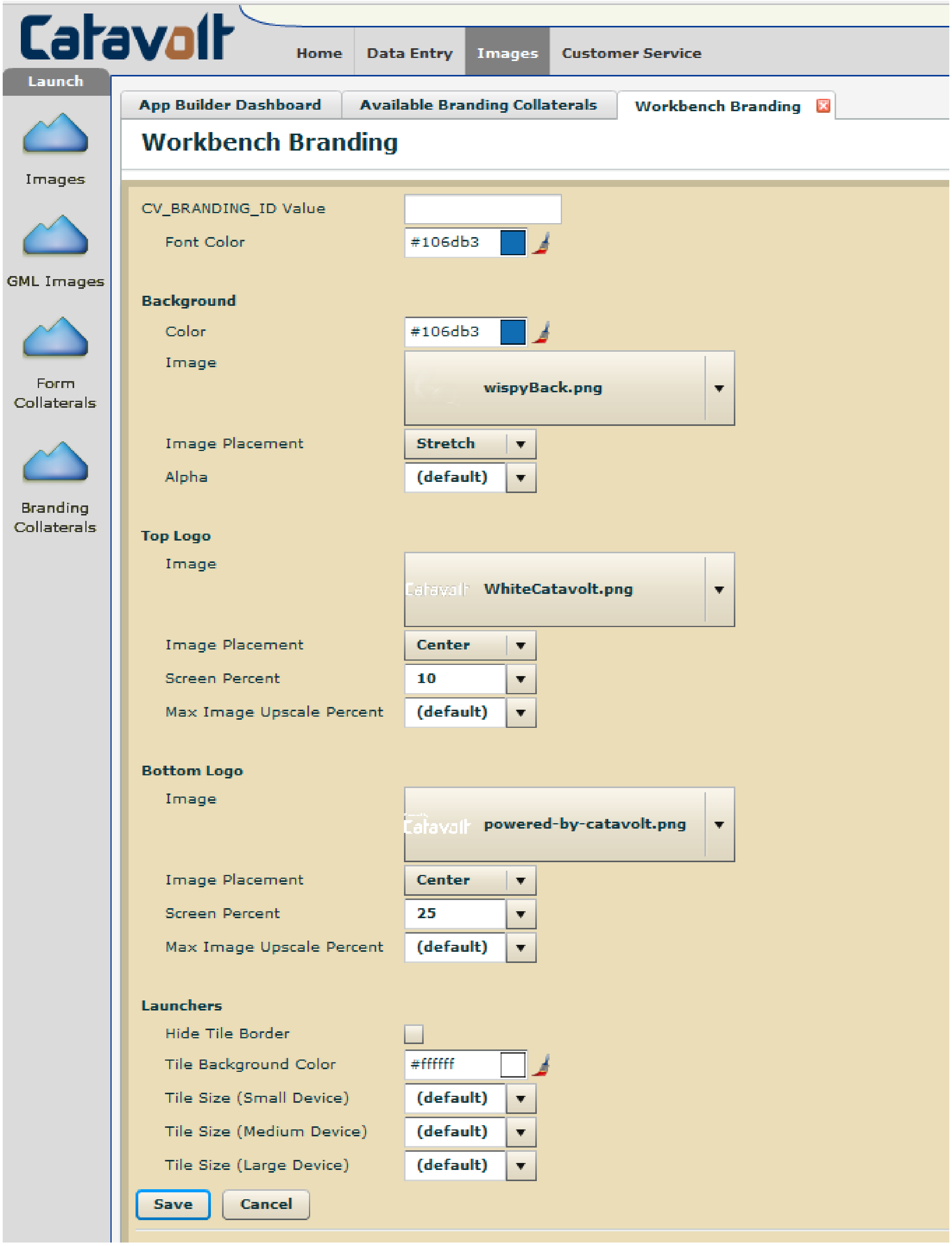
Bottom Logo Image Placement specifies where to place the bottom logo on the login prompt. Leaving this value blank will use the default placement. Allowable values are:

- Top – Place directly below the login credentials (user / password), putting any extra space below.
- Center – Center the image in the bottom logo area, splitting any extra space above and below.
- Bottom – Place directly at the bottom of the screen, putting any extra space above.

Bottom Logo Screen Percent specifies the vertical percent (1-100) of the screen that the bottom logo should occupy. Leaving this value blank will use the default percentage. This value is typically set from 10 to 40 percent.

Bottom Logo Max Upscale Percent specifies the max percentage (101-9999) that the image can be upscaled in order to fit on the screen. Leaving this value blank will keep the image at 100% (no upscaling).

If you select Workbench, you will be presented with the following dialog:



Workbench Branding

CV_BRANDING_ID Value

Font Color #106db3

Background

Color #106db3

Image wispyBack.png

Image Placement Stretch

Alpha (default)

Top Logo

Image WhiteCatavolt.png

Image Placement Center

Screen Percent 10

Max Image Upscale Percent (default)

Bottom Logo

Image powered-by-catavolt.png

Image Placement Center

Screen Percent 25

Max Image Upscale Percent (default)

Launchers

Hide Tile Border □

Tile Background Color #ffffff

Tile Size (Small Device) (default)

Tile Size (Medium Device) (default)

Tile Size (Large Device) (default)

Figure 14: Create Workbench Branding File dialog

CV_BRANDING_ID Value allows you to specify an alternate branding ID. If you leave this value blank, a brandingLauncher.json file will be created that will specify the Workbench page branding for all users. If you want to provide alternate branding for specific users or classes of user, specify a value for this property. A brandingLauncher-<value>.json file will be created to specify the Workbench page branding (see note below).

Font Color specifies the font color for the labels of the workbench page. Leaving this value blank will use the default color.

Background Color specifies the background color of the workbench page. Leaving this value blank will use the default color.

Background Image specifies an image to be used as the background of the workbench page. Leaving this value blank will use no image.

Background Image Placement specifies where to place the background image of the workbench page. Leaving this value blank will use the default placement. Allowable values are:

- Stretch – Stretch the image to fill the screen.
- Tile – Tile multiple copies of the image as needed to fill the screen.
- Fit – Adjust the image to fit the screen.

Background Alpha specifies a transparency value between 0 and 1 for the background image. A value of 0.0 indicates full transparency. A value of 1.0 indicates no transparency. Leaving this value blank will use no transparency.

Top Logo Image specifies an image to be used at the top of the workbench page. Leaving this value blank will use no image.

Top Logo Image Placement specifies where to place the bottom logo of the workbench page. Leaving this value blank will use the default placement. Allowable values are:

- Left – Place on the left of the screen.
- Center – Place in the center of the screen.
- Right – Place on the right of the screen.

Top Logo Screen Percent specifies the vertical percent (1-100) of the screen that the top logo should occupy. Leaving this value blank will use the default percentage. This value is typically set from 20 to 40 percent.

Top Logo Max Upscale Percent specifies the max percentage (101-9999) that the image can be upscaled in order to fit on the screen. Leaving this value blank will keep the image at 100% (no upscaling).

Bottom Logo Image specifies an image to be used at the bottom of the workbench page. Leaving this value blank will use no image.

Bottom Logo Image Placement specifies where to place the bottom logo of the workbench page. Leaving this value blank will use the default placement. Allowable values are:

- Left – Place on the left of the screen.
- Center – Place in the center of the screen.
- Right – Place on the right of the screen.

Bottom Logo Screen Percent specifies the vertical percent (1-100) of the screen that the bottom logo should occupy. Leaving this value blank will use the default percentage. This value is typically set from 10 to 40 percent.

Bottom Logo Max Upscale Percent specifies the max percentage (101-9999) that the image can be upscaled in order to fit on the screen. Leaving this value blank will keep the image at 100% (no upscaling).

Hide Tile Border specifies to not draw a border around the workbench launcher image, effectively making the image itself be the border of the launcher.

Tile Background Color specifies the background color of the workbench tile. Leaving this value blank will use the default color.

Tile Size (small device) specifies a uniform pixel-density independent size (1-9999) for tiles on the workbench dialog for a small device (phone). Leaving this value blank will cause tiles to be sized based on the largest workbench launcher image.

Tile Size (medium device) specifies a uniform pixel-density independent size (1-9999) for tiles on the workbench dialog for a medium device (small tablet). Leaving this value blank will cause tiles to be sized based on the largest workbench launcher image.

Tile Size (large device) specifies a uniform pixel-density independent size (1-9999) for tiles on the workbench dialog for a large device (tablet). Leaving this value blank will cause tiles to be sized based on the largest workbench launcher image.

A Note on CV_BRANDING_ID

For a majority of companies, one set of branding is sufficient for all users. In this case, you leave CV_BRANDING_ID Value blank at all times.

Some companies may wish to show different brandings for each user or group of users. To accomplish this, create a CV_BRANDING_ID User Property for each user that needs the alternate branding. For example, for user Bob Jones, create a CV_BRANDING_ID User Property with the value of ABC. Then, create branding files above with a CV_BRANDING_ID Value of ABC (case is sensitive). When Bob Jones logs in, he will see the alternate branding you created, while other users will see the standard branding.

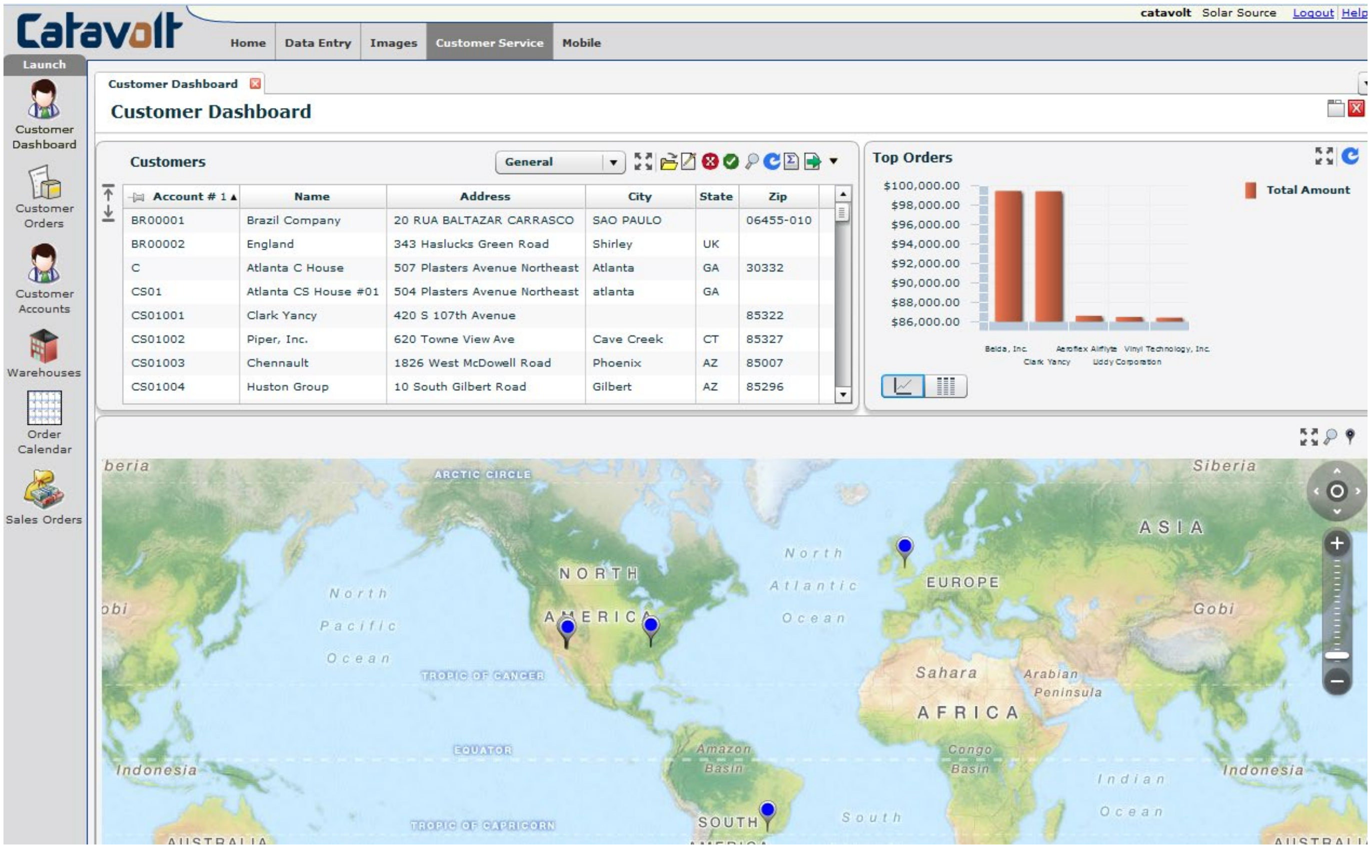
Changing a Branding File

To change a Branding file, select it from the list and choose the Edit action. Note this option is for non-image files. To change an image, simply upload a new version of the image with the same file name.

Chapter 14: Dashboards

Chapter Summary

When defining a Data Object Detail, Xalt | Mobility allows you to add Queries (lists) of other objects as sections in your Detail. You may have cases where you would like to create a Dashboard of multiple Queries without having an obvious Detail. For example, you may want to show a list of Customers with a Customer Map and a graph showing Top 5 Orders all on the same window. You can now do this more easily by creating a Dashboard.

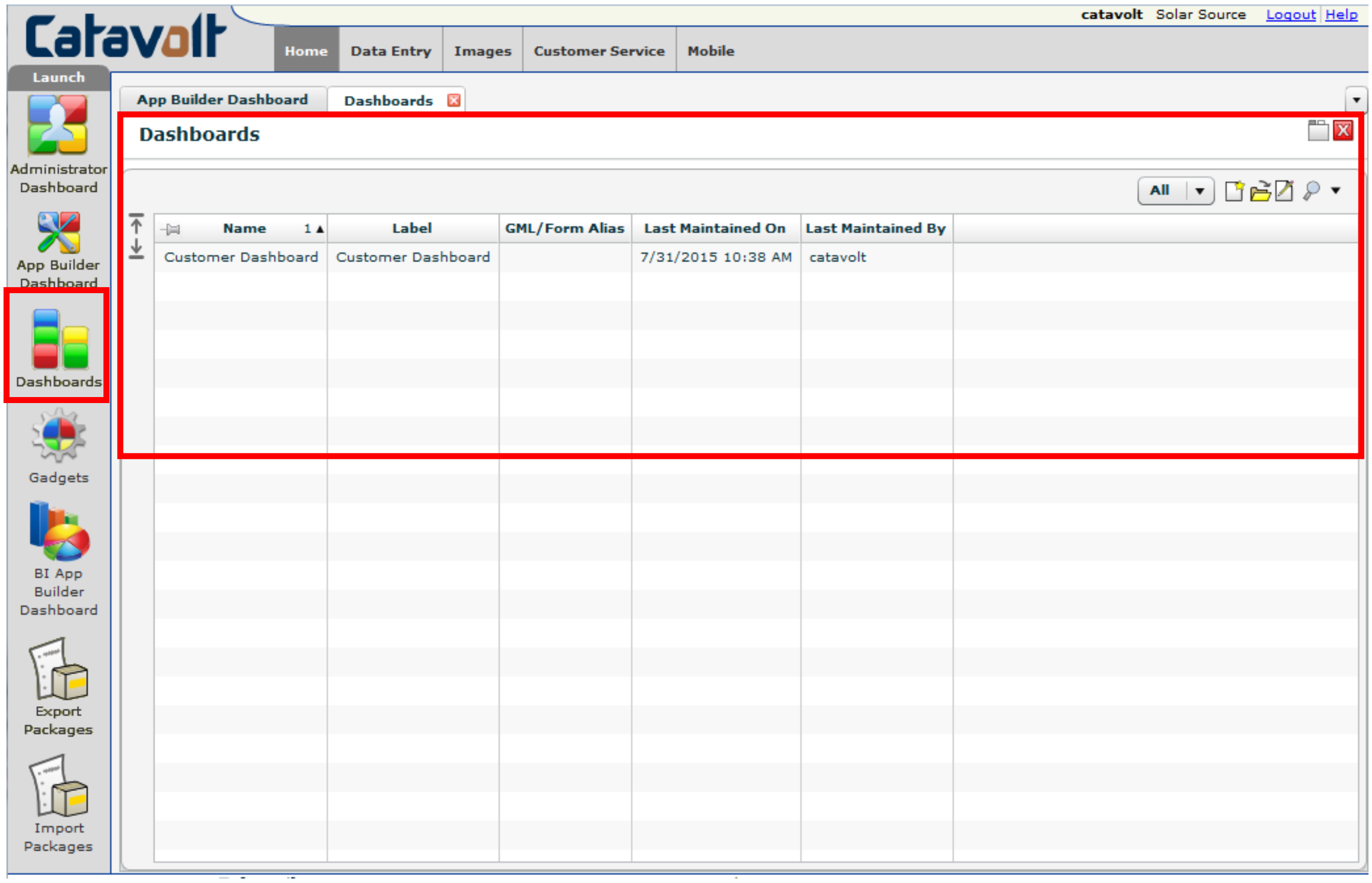


Account # 1 ▲	Name	Address	City	State	Zip
BR00001	Brazil Company	20 RUA BALTAZAR CARRASCO	SAO PAULO		06455-010
BR00002	England	343 Haslucks Green Road	Shirley	UK	
C	Atlanta C House	507 Plasters Avenue Northeast	Atlanta	GA	30332
CS01	Atlanta CS House #01	504 Plasters Avenue Northeast	atlanta	GA	
CS01001	Clark Yancy	420 S 107th Avenue			85322
CS01002	Piper, Inc.	620 Towne View Ave	Cave Creek	CT	85327
CS01003	Chennault	1826 West McDowell Road	Phoenix	AZ	85007
CS01004	Huston Group	10 South Gilbert Road	Gilbert	AZ	85296

Figure 1: An example Customer Dashboard with 3 unrelated sections.

Accessing Dashboards

Dashboards can be accessed by clicking on the Dashboards Launcher on the Xalt Home Workbench.



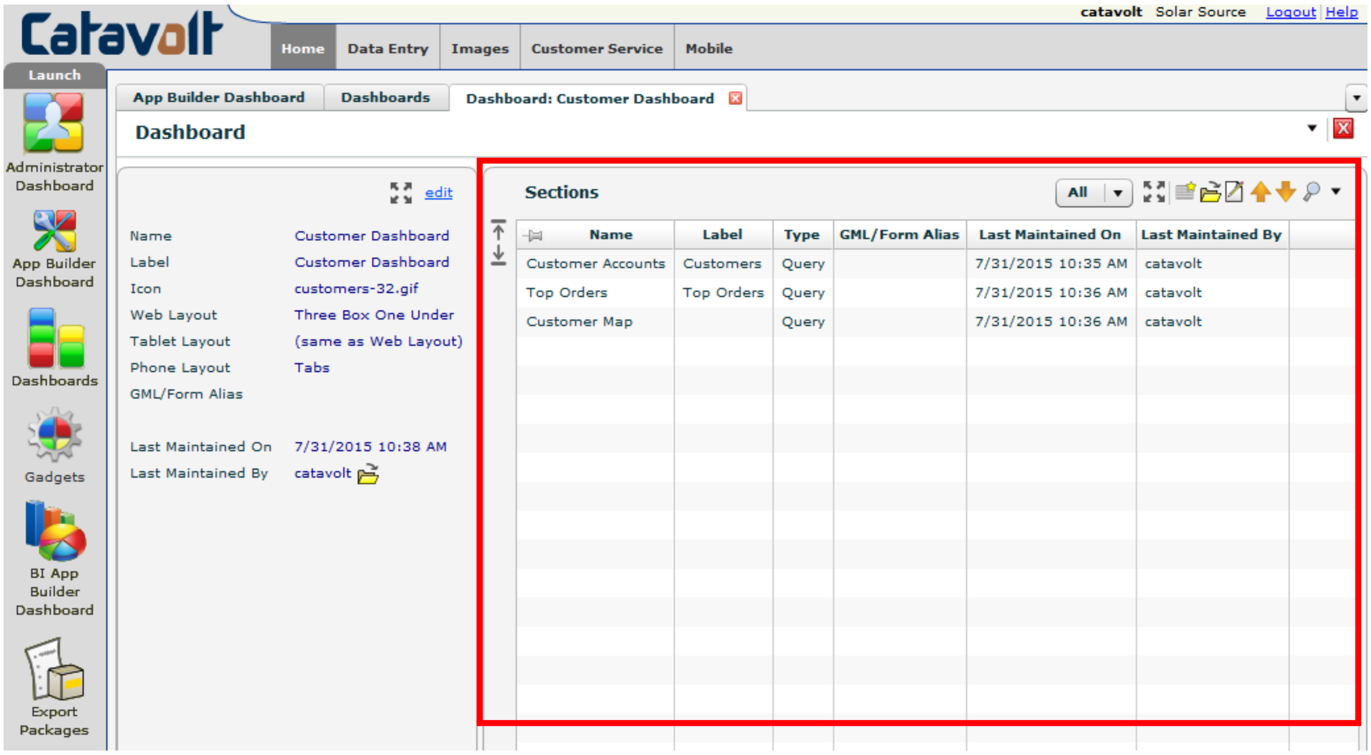
Dashboards

Name	Label	GML/Form Alias	Last Maintained On	Last Maintained By
Customer Dashboard	Customer Dashboard		7/31/2015 10:38 AM	catavolt

Figure 2: The administrator's Home workbench showing the Dashboards launcher with the Dashboards list view open.

Dashboard Components

When you display the definition for a Dashboard, there are 2 Details that you can select from. The Default Detail shows you all the information about the Dashboard. A Dashboard has 1 component: Sections (a list of sections to be displayed in the Dashboard).



Dashboard

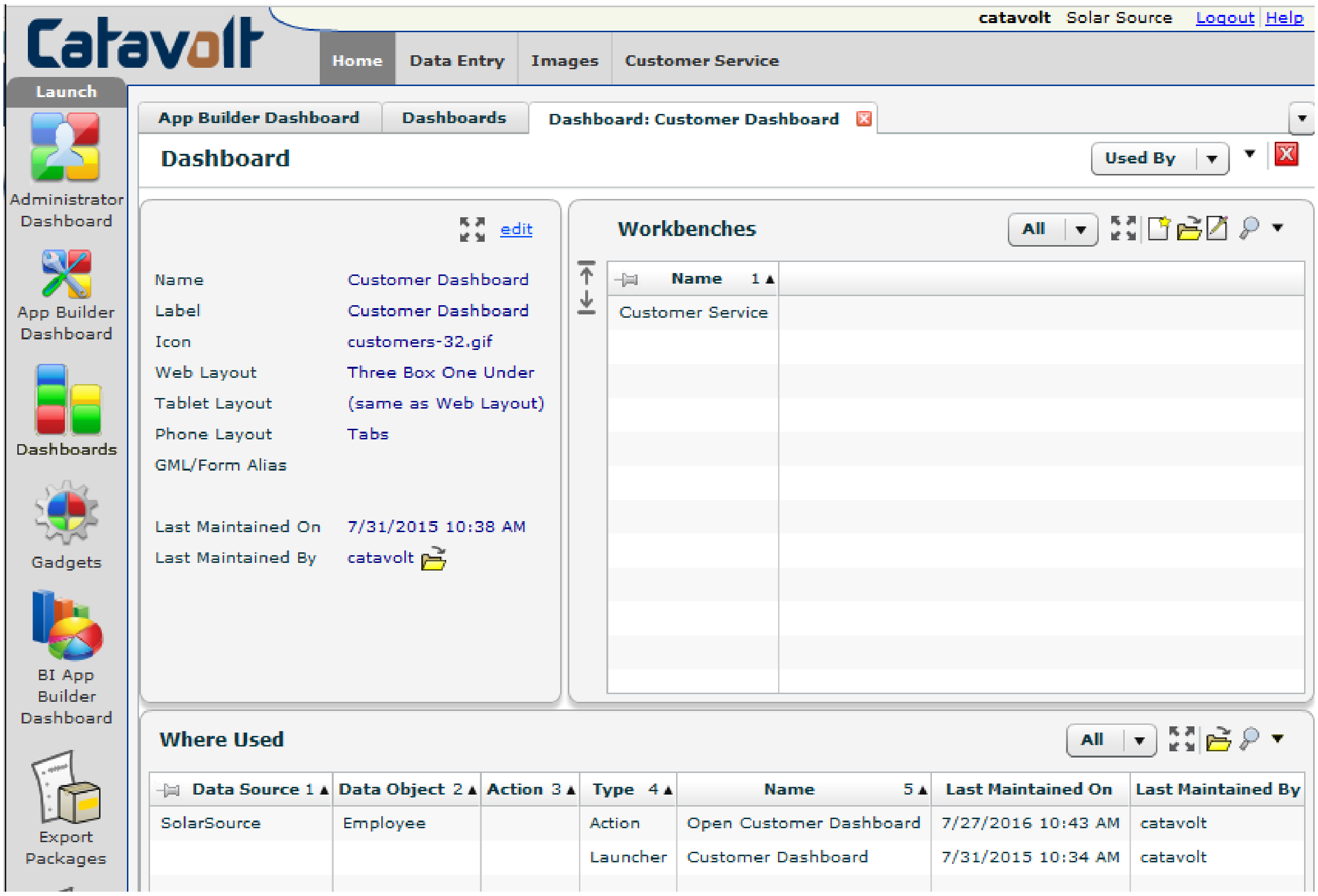
Name	Customer Dashboard
Label	Customer Dashboard
Icon	customers-32.gif
Web Layout	Three Box One Under
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Last Maintained On	7/31/2015 10:38 AM
Last Maintained By	catavolt

Sections

Name	Label	Type	GML/Form Alias	Last Maintained On	Last Maintained By
Customer Accounts	Customers	Query		7/31/2015 10:35 AM	catavolt
Top Orders	Top Orders	Query		7/31/2015 10:36 AM	catavolt
Customer Map		Query		7/31/2015 10:36 AM	catavolt

Figure 3: The Dashboard details view with the Sections query section highlighted.

The Used By Detail shows a list of all Workbenches that currently have this Dashboard on them as well as a Where Used list of objects that are currently using this Dashboard.



Name	Customer Dashboard
Label	Customer Dashboard
Icon	customers-32.gif
Web Layout	Three Box One Under
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Last Maintained On	7/31/2015 10:38 AM
Last Maintained By	catavolt

Workbenches

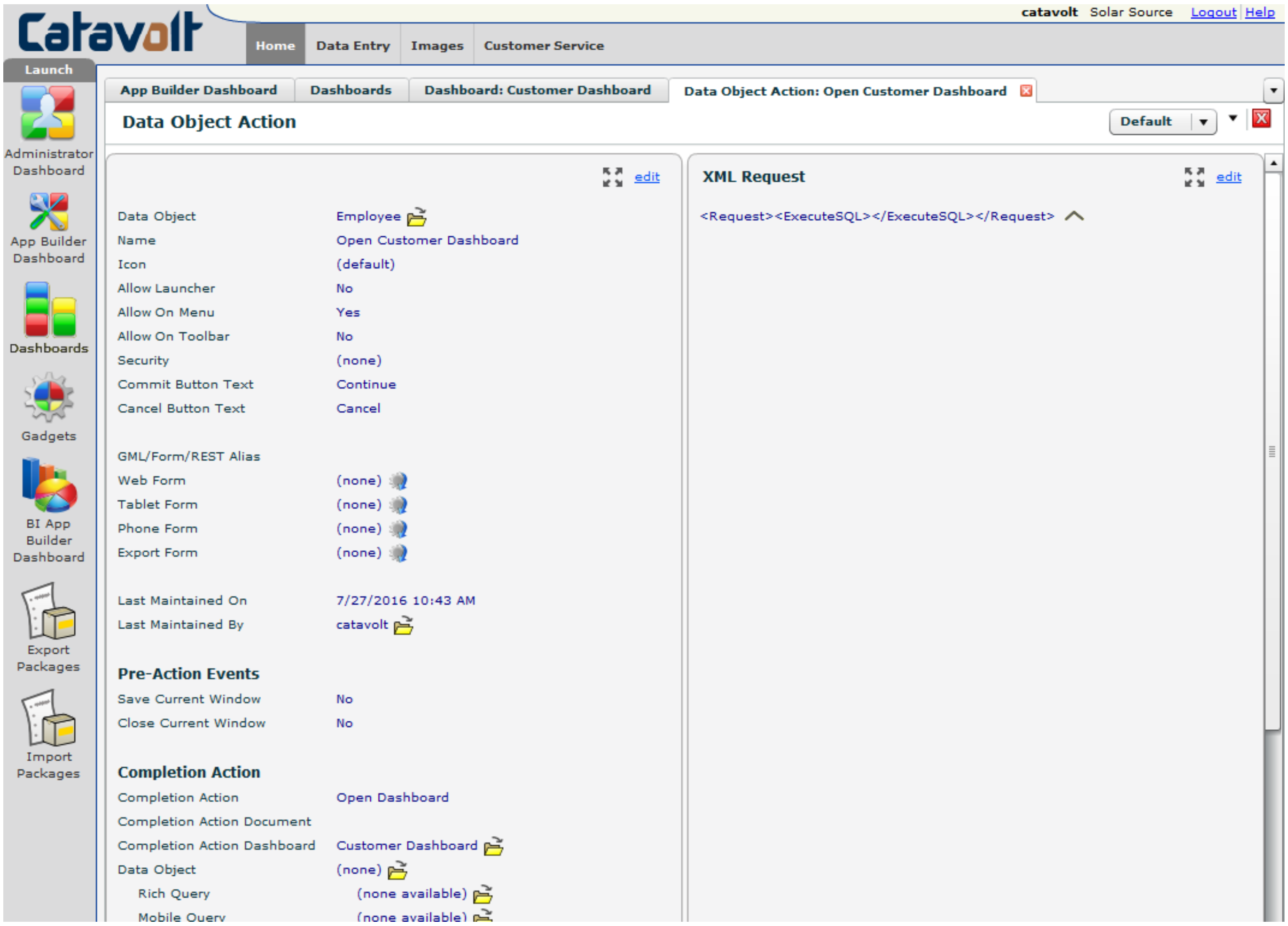
Name 1 ▲
Customer Service

Where Used

Data Source 1 ▲	Data Object 2 ▲	Action 3 ▲	Type 4 ▲	Name 5 ▲	Last Maintained On	Last Maintained By
SolarSource	Employee		Action	Open Customer Dashboard	7/27/2016 10:43 AM	catavolt
			Launcher	Customer Dashboard	7/31/2015 10:34 AM	catavolt

Figure 4: Dashboard 'Used By' details

Opening a Where Used record will take you directly to the object using the Dashboard:



Data Object Action

Data Object	Employee
Name	Open Customer Dashboard
Icon	(default)
Allow Launcher	No
Allow On Menu	Yes
Allow On Toolbar	No
Security	(none)
Commit Button Text	Continue
Cancel Button Text	Cancel
GML/Form/REST Alias	
Web Form	(none)
Tablet Form	(none)
Phone Form	(none)
Export Form	(none)
Last Maintained On	7/27/2016 10:43 AM
Last Maintained By	catavolt

Pre-Action Events

Save Current Window	No
Close Current Window	No

Completion Action

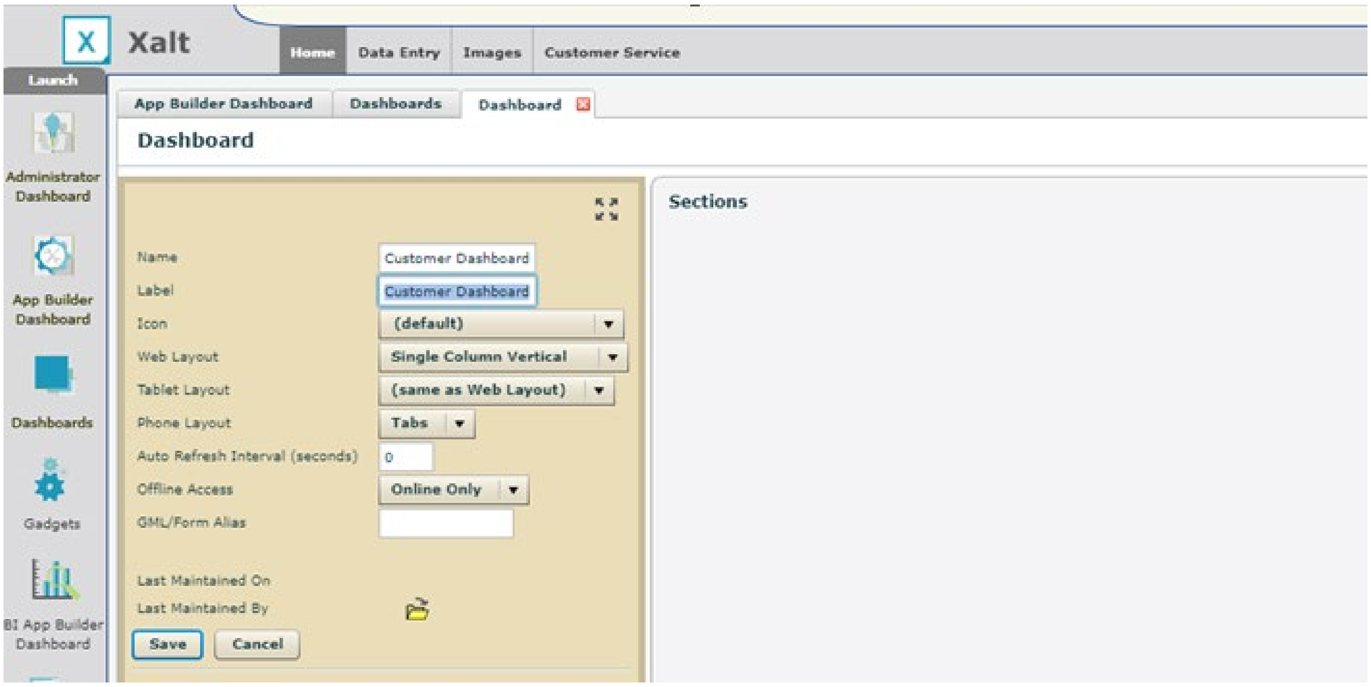
Completion Action	Open Dashboard
Completion Action Document	
Completion Action Dashboard	Customer Dashboard
Data Object	(none)
Rich Query	(none available)
Mobile Query	(none available)

XML Request

```
<Request><ExecuteSQL></ExecuteSQL></Request>
```

Figure 5: Dashboard Where Used

Creating Dashboards



Dashboard

Name: Customer Dashboard
Label: Customer Dashboard
Icon: (default)
Web Layout: Single Column Vertical
Tablet Layout: (same as Web Layout)
Phone Layout: Tabs
Auto Refresh Interval (seconds): 0
Offline Access: Online Only
GML/Form Alias:
Last Maintained On
Last Maintained By

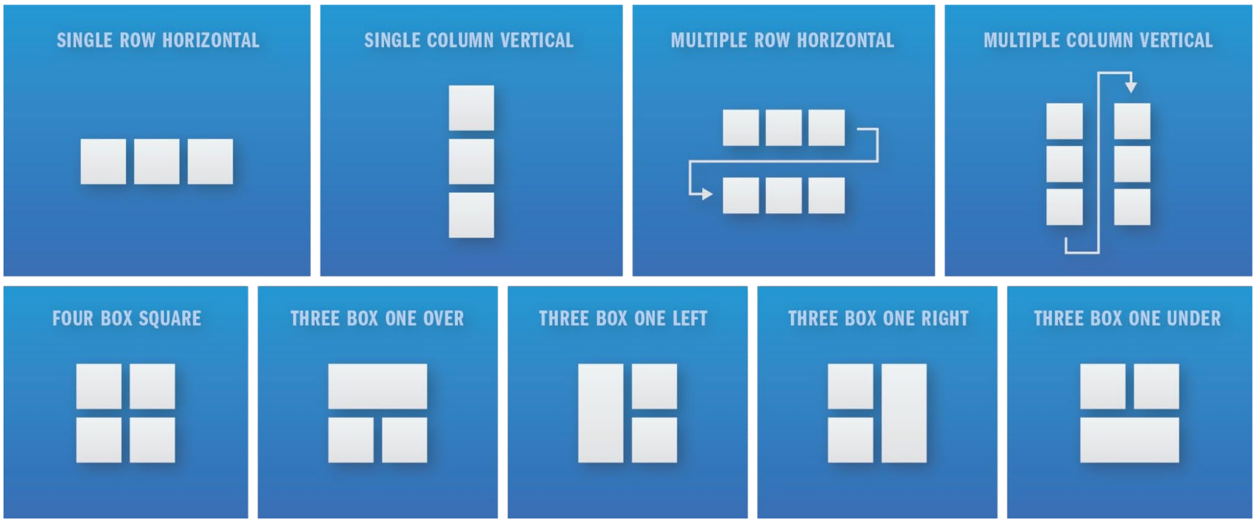
Figure 6: The create Dashboard details view.

When creating a Dashboard, you must specify the Name. This value is used to uniquely identify the Dashboard and is used in the dashboard drop down at runtime.

Label specifies an identifying description of the section as displayed to the end user in the top left corner. Leaving this value empty will cause no label to be displayed for the section.

Icon specifies the image that will appear on the Launcher.

Web Layout specifies how you want to arrange the different sections when viewed by a Web Browser. The following options are available:



- Single Row Horizontal – All sections are arrange side-by-side in a single row
- Single Column Vertical – All sections are arrange on top of each other in a single column

- Multiple Row Horizontal – Sections are arranged side-by-side. Sections fill up a row based on the window size and then wrap to the next row as needed. As the window is resized the sections adjust and wrap. This is similar to word wrapping in word processing applications.
- Multiple Column Vertical – Sections are arranged on top of each other. Sections fill up a column based on the window size and then wrap to the top of the next column. As the window is resized the sections adjust and wrap.
- Four Box Square – Up to 4 sections are equally sized and displayed in a 2x2 area. Any sections beyond the fourth are added underneath using the "Single Column Vertical" style.
- Three Box One Over – Up to 3 sections are placed, the first on row one and the other two side-by-side on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Left – Up to 3 sections are placed, the first on column one and the other two on top of each other on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Right – Up to 3 sections are placed, the first two on top of each other on column one and the third on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Under – Up to 3 sections are placed, the first two side-by-side on row one and the third on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Tabs – Each section is represented as a Tab with the detail. Each tab fills the entire detail section.
- Maintainable Query – Sections are arranged in Single Column Vertical layout with special Maintainable Query options. See Maintainable Query section below for more information on Maintainable Queries.

Tablet Layout specifies how you want to arrange the different sections when viewed on a Tablet-size mobile device. The following options are available:

- (same as web layout) – Use the same layout as the rich client, if it is available on mobile. If the web layout is not available on the mobile device, Single Column Vertical will be used.
- Single Column Vertical – All sections are arrange on top of each other in a single column
- Four Box Square – Up to 4 sections are equally sized and displayed in a 2x2 area. Any sections beyond the fourth are added underneath using the "Single Column Vertical" style.
- Three Box One Over – Up to 3 sections are placed, the first on row one and the other two side-by-side on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Left – Up to 3 sections are placed, the first on column one and the other two on top of each other on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Right – Up to 3 sections are placed, the first two on top of each other on column one and the third on column two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Three Box One Under – Up to 3 sections are placed, the first two side-by-side on row one and the third on row two. Any sections beyond the third are added underneath using the "Single Column Vertical" style.
- Tabs - Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

Phone Layout specifies how you want to arrange the different sections when viewed on a Phone-size mobile device. The following options are available:

- Tabs - Each section is represented as a Tab with the detail. Each tab fills the entire detail section.

Auto Refresh Interval (seconds) specifies the amount of time that should pass before the Dashboard window automatically refreshes itself. Allowable values are 0 (no auto-refresh) or between 5 and 14400 seconds (4 hours). The default value is 0 (no auto-refresh). Turning on auto-refresh has the potential to greatly increase the load on your back-end server depending on the time set and the number of users accessing this Dashboard. When Auto Refresh is applied to a Dashboard, the setting is used for the entire Dashboard window, including all sub-sections. If the Dashboard contains Query Sections that point to other Data Objects that also have auto-refresh enabled, the refresh timers for those Data Objects are ignored. The refresh timers for those Data Object will be applied again when drilling into Details for those Data Objects.

Offline Access allows you to specify whether this Dashboard can be shown while offline. The available options are:

Offline – Take this Dashboard offline

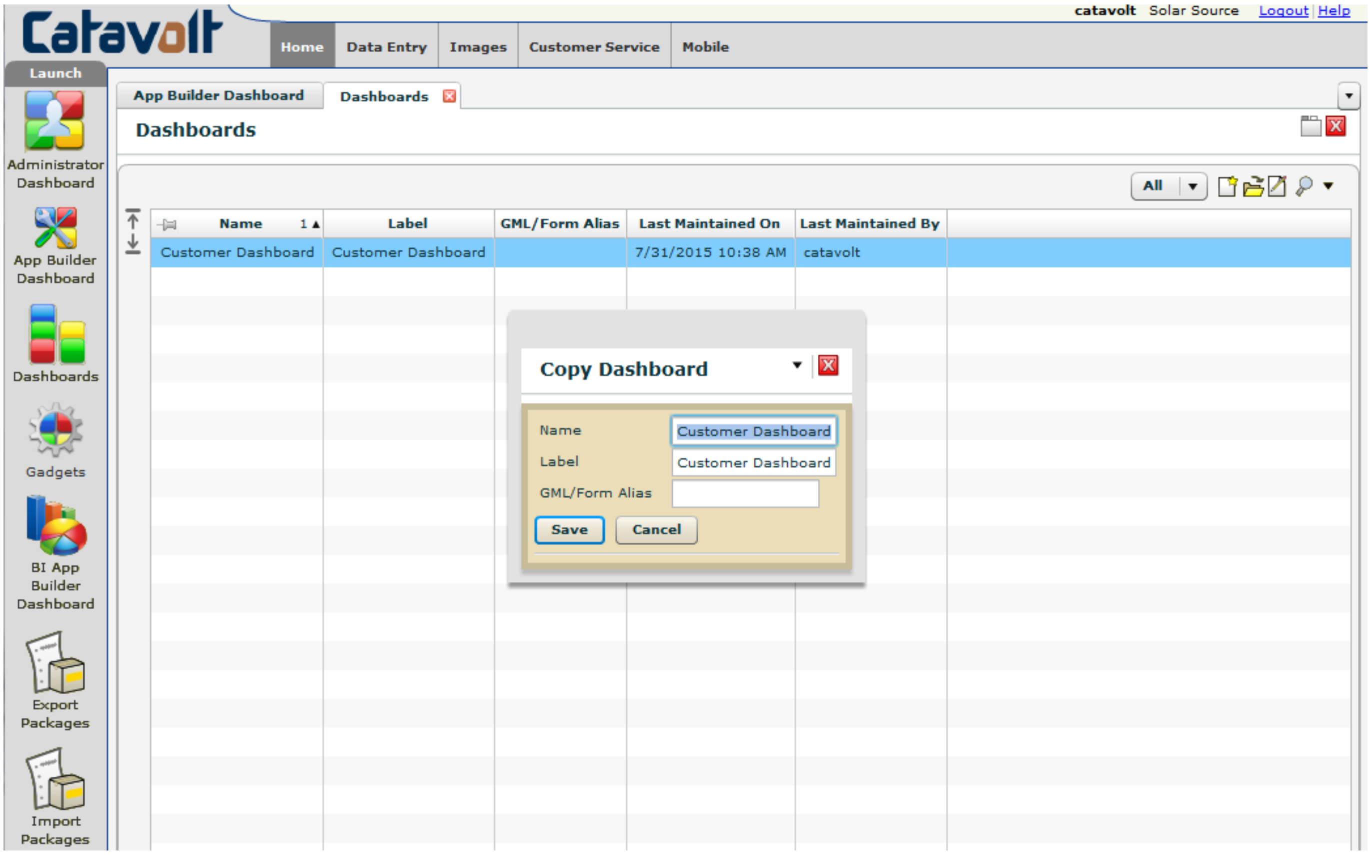
Online Only – Do not allow this Dashboard to be taken offline

Please see the **Xalt Mobility – Offline Guide** for more information about Offline Access.

GML/Form Alias allows you to specify an alias when using this Dashboard with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Copying Dashboards

You may have instances where you need to make a copy of an existing Dashboard to show a different set of sections. You can select the Copy menu option to accomplish this. When Copying a Dashboard, you will be prompted to supply the new Name. You can also choose to change the Label and GML/Form Alias. An exact copy of this Dashboard along with all of its components will be made.



Dashboards

Name 1 ▲	Label	GML/Form Alias	Last Maintained On	Last Maintained By
Customer Dashboard	Customer Dashboard		7/31/2015 10:38 AM	catavolt

Copy Dashboard

Name: Customer Dashboard

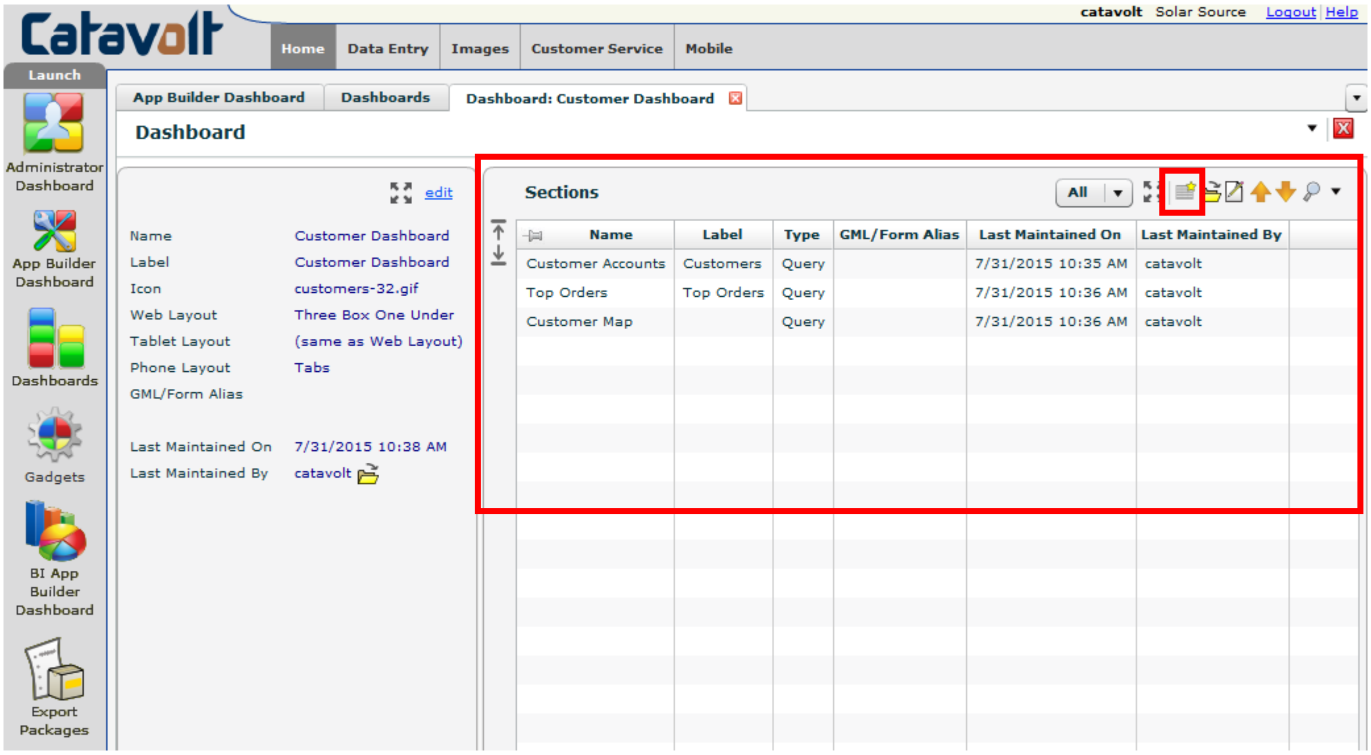
Label: Customer Dashboard

GML/Form Alias:

Figure 7: The Dashboard copy prompt

Dashboard Sections

The Sections section shows a list of all sections defined for the specified dashboard. There is only one kind of Section: Query Sections. (Note: If you need a Property Section, you can already use a Detail with a Detail Launcher to accomplish this without a Dashboard).



Dashboard

Name	Customer Dashboard
Label	Customer Dashboard
Icon	customers-32.gif
Web Layout	Three Box One Under
Tablet Layout	(same as Web Layout)
Phone Layout	Tabs
GML/Form Alias	
Last Maintained On	7/31/2015 10:38 AM
Last Maintained By	catavolt

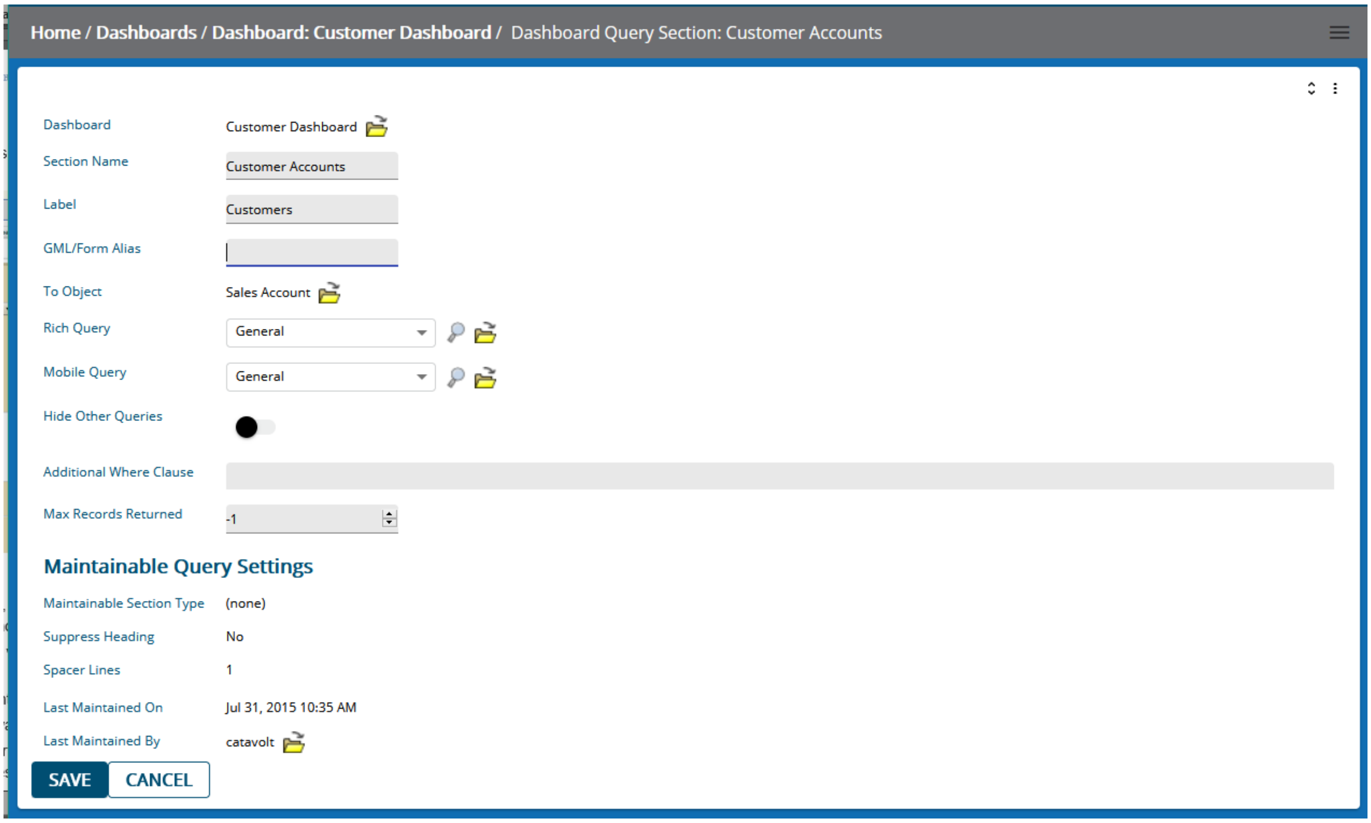
Sections

Name	Label	Type	GML/Form Alias	Last Maintained On	Last Maintained By
Customer Accounts	Customers	Query		7/31/2015 10:35 AM	catavolt
Top Orders	Top Orders	Query		7/31/2015 10:36 AM	catavolt
Customer Map		Query		7/31/2015 10:36 AM	catavolt

Figure 8: Dashboard details view with the Sections query section highlighted along with the Add Query Section toolbar actions

Query Section

A Query Section will display a list of records in a section for the given Data Object.



Home / Dashboards / Dashboard: Customer Dashboard / Dashboard Query Section: Customer Accounts

Dashboard	Customer Dashboard
Section Name	Customer Accounts
Label	Customers
GML/Form Alias	
To Object	Sales Account
Rich Query	General
Mobile Query	General
Hide Other Queries	
Additional Where Clause	
Max Records Returned	-1

Maintainable Query Settings

Maintainable Section Type	(none)
Suppress Heading	No
Spacer Lines	1
Last Maintained On	Jul 31, 2015 10:35 AM
Last Maintained By	catavolt

Figure 9: Dashboard Query Section details view

When creating a Dashboard Query Section, you must specify the Section Name. This value is used to uniquely identify the Query Section and is not presented to the end user.

To Object specifies which Data Object you will display a list for. This list will contain all Data Objects defined for all Data Sources.

Rich Query specifies which Query you want to use when displaying this section on a Rich client.

Mobile Query specifies which Query you want to use when displaying this section on a Mobile client.

Hide Other Queries specifies whether the runtime Query dropdown list will contain just the selected Rich/Mobile Query or all non-hidden Queries. If Hide Other Queries is selected, only the selected Rich/Mobile Query will be available to choose in the Query dropdown list.

Section Name is used to uniquely identify the Property Section and is not presented to the end user. Xalt defaults this value to the Object Name (Plural) of the selected To Object.

Label specifies an identifying description of the section as displayed to the end user in the top left corner. Leaving this value empty will cause no label to be displayed for the section. Xalt defaults this value to the Object Name (Plural) of the selected To Object.

GML/Form Alias allows you to specify an alias when using this Section with GML and Custom Forms. Your Hexagon sales representative has more information about using GML/Form aliases.

Additional Where Clause specifies additional criteria that can be used to further filter the list of records to be displayed. If you do not specify an Additional Where Clause, you will get an unfiltered list, for example all order line items for all orders. You may use Session/User/Default substitution values in the where clause. See Appendix B: Specifying Where Clauses for more information about specifying Additional Where Clause values.

Max Records Returned specifies the number of records to return in the list. Xalt defaults this value to -1, which will return all records.

The Maintainable Query Settings section contains values that are used when this Query Section is used as part of a Maintainable Query Dashboard. See Maintainable Query section below for more information on Maintainable Queries.:

Maintainable Section Type specifies the type of section. The available values are:

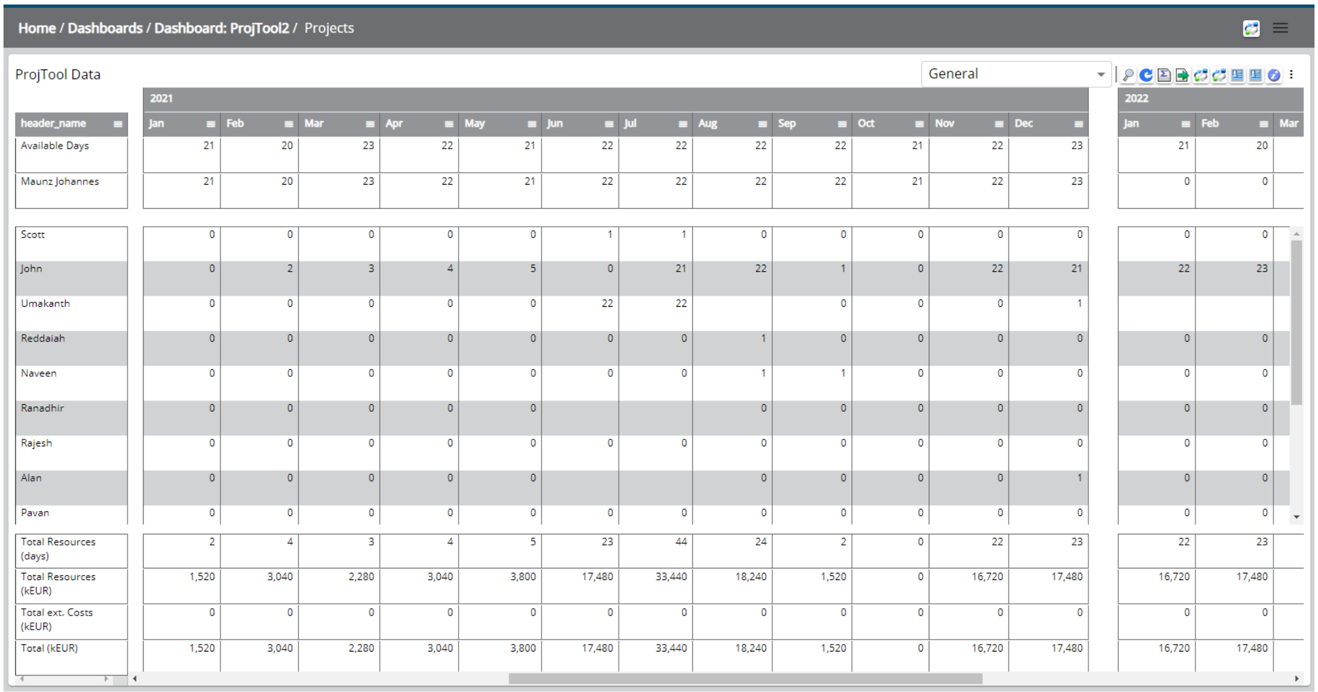
- Header – Read-only section to be displayed first (at the top). You can specify multiple header sections in a Maintainable Query.
- Data – Maintainable section to be displayed between the header and footer section(s). You can only specify 1 data section in a Maintainable Query.
- Footer – Read-only section to be displayed last (at the bottom). You can specify multiple footer sections in a Maintainable Query.

Suppress Heading specifies whether to hide or show the heading information for the Query section. Selecting this option will hide all column headings, column groups, and the sections title area (title, menu, query dropdown list).

Spacer Lines specifies the number of spacer lines to include between this section and the section following this one (default is 1). If set to 0, the 2 sections will be placed together with no intervening spaces. If Suppress Heading is also specified on the section after, the 2 sections will appear to the user as a single table.

Maintainable Queries

A new Web Layout (available with XHA only) called **Maintainable Query** has been added to allow you to create a Dashboard with multiple sections that appear as a single section.



Home / Dashboards / Dashboard: ProjTool2 / Projects

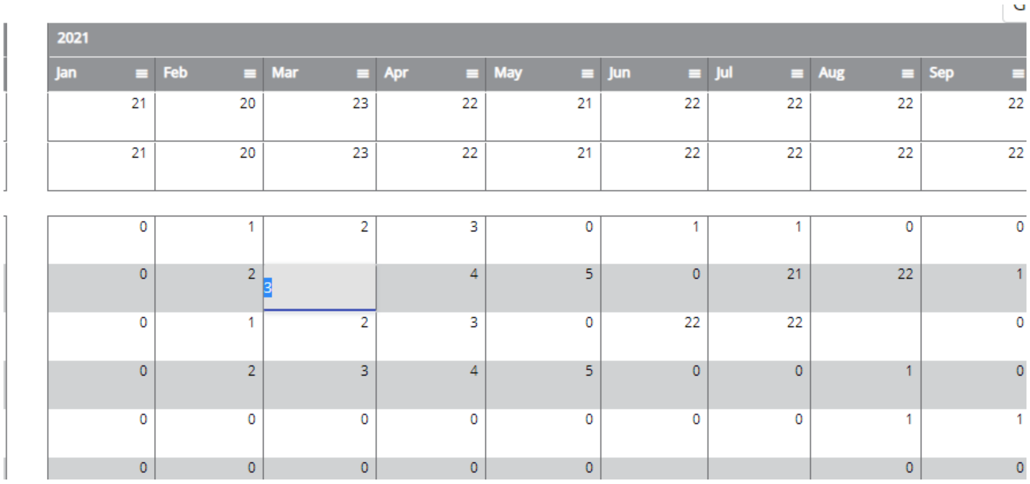
ProjTool Data

General

	2021												2022		
header_name	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar
Available Days	21	20	23	22	21	22	22	22	22	21	22	23	21	20	
Maunz Johannes	21	20	23	22	21	22	22	22	22	21	22	23	0	0	
Scott	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
John	0	2	3	4	5	0	21	22	1	0	22	21	22	23	
Umakanth	0	0	0	0	0	22	22		0	0	0	1			
Reddaiah	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
Naveen	0	0	0	0	0	0	0	1	1	0	0	0	0	0	
Ranadhir	0	0	0	0	0			0	0	0	0	0	0	0	
Rajesh	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Alan	0	0	0	0	0			0	0	0	0	1	0	0	
Pavan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total Resources (days)	2	4	3	4	5	23	44	24	2	0	22	23	22	23	
Total Resources (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	
Total ext. Costs (kEUR)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	

Figure 10: Maintainable Query Dashboard runtime view

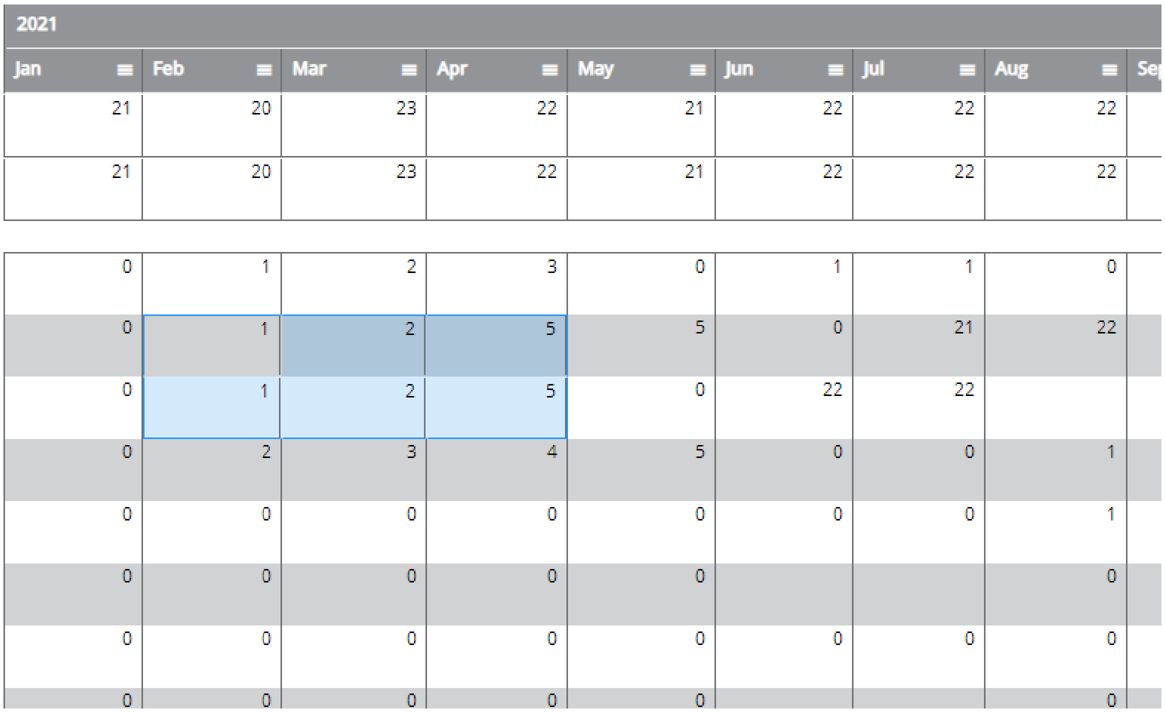
Records in the **Data** section can be maintained directly from the list without the need to open a separate Detail window. Double-clicking a cell will allow you to change the value inside the cell (if the Query View Property for that column is set to Maintainable). You can press Tab and Shift+Tab to move to the next/previous cell or click into another cell in the same row to enter more values. Tabbing past the last column in a row, selecting another row, or pressing the Enter key will submit all changed values in the row for update.



2021								
Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep
21	20	23	22	21	22	22	22	22
21	20	23	22	21	22	22	22	22
0	1	2	3	0	1	1	0	0
0	2	3	4	5	0	21	22	1
0	1	2	3	0	22	22		0
0	2	3	4	5	0	0	1	0
0	0	0	0	0	0	0	1	1
0	0	0	0	0			0	0

Figure 11: Maintainable Query Dashboard with a cell in edit mode

You can select single or multiple rows/columns and use copy/paste to transfer these values to other rows/columns, similar to a spreadsheet:



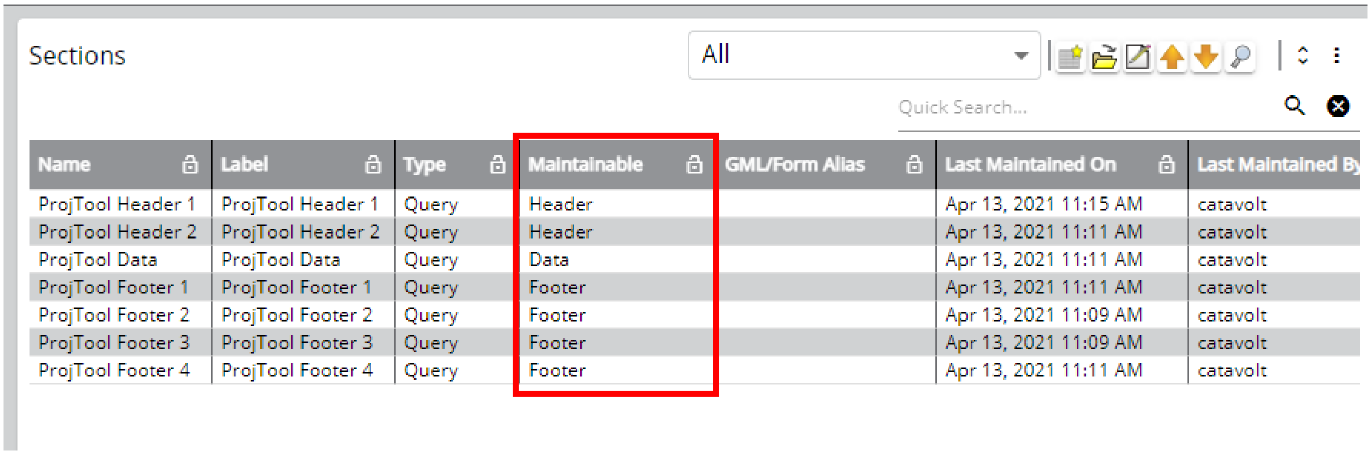
2021								
Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Se
21	20	23	22	21	22	22	22	
21	20	23	22	21	22	22	22	
0	1	2	3	0	1	1	0	
0	1	2	5	5	0	21	22	
0	1	2	5	0	22	22		
0	2	3	4	5	0	0	1	
0	0	0	0	0	0	0	1	
0	0	0	0	0			0	
0	0	0	0	0	0	0	0	
0	0	0	0	0			0	

Figure 12: Maintainable Query Dashboard runtime view with multiple rows/columns selected

All changed records will be updated at the time the paste occurs. Any rows that failed to update will be highlighted in red and error messages will be presented to the user. The user can then manually fix each failing row and resubmit each update, or they can cancel update on all failing rows.

Designing Maintainable Queries

Maintainable Query Sections are laid out as Single Column Vertical and are defined as Header, Data, and Footer:

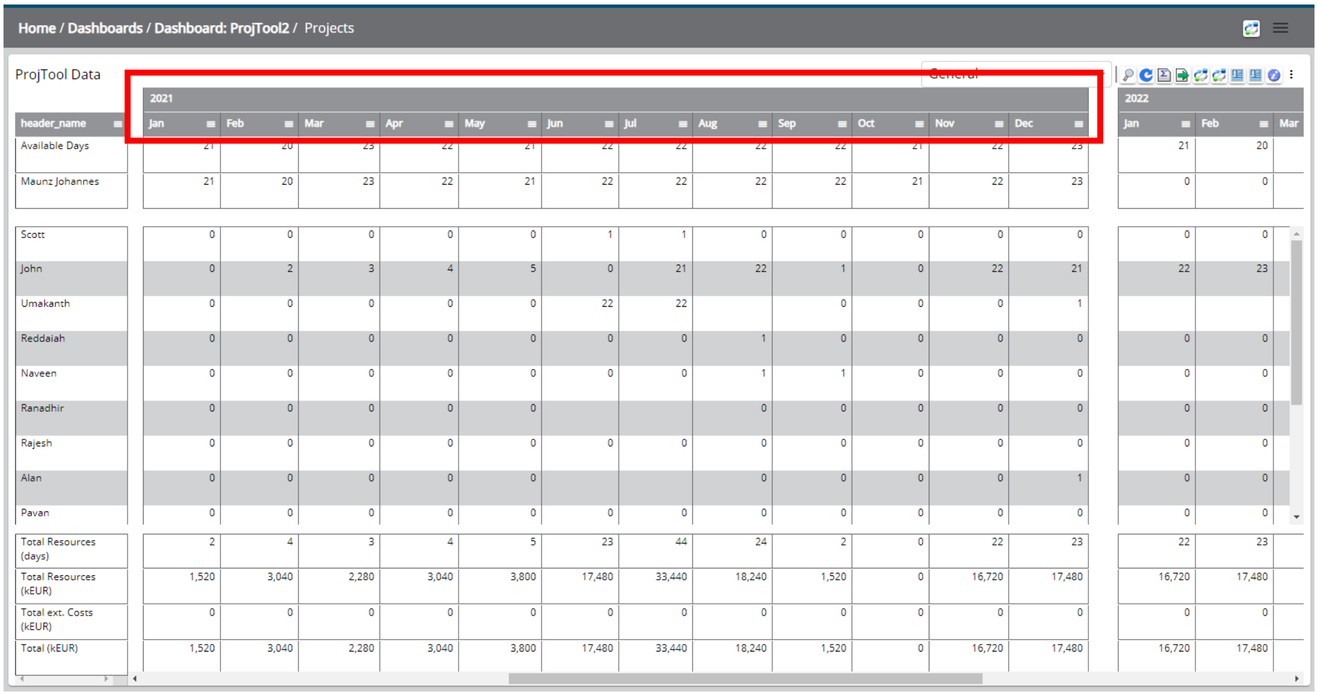


Name	Label	Type	Maintainable	GML/Form Alias	Last Maintained On	Last Maintained By
ProjTool Header 1	ProjTool Header 1	Query	Header		Apr 13, 2021 11:15 AM	catavolt
ProjTool Header 2	ProjTool Header 2	Query	Header		Apr 13, 2021 11:11 AM	catavolt
ProjTool Data	ProjTool Data	Query	Data		Apr 13, 2021 11:11 AM	catavolt
ProjTool Footer 1	ProjTool Footer 1	Query	Footer		Apr 13, 2021 11:11 AM	catavolt
ProjTool Footer 2	ProjTool Footer 2	Query	Footer		Apr 13, 2021 11:09 AM	catavolt
ProjTool Footer 3	ProjTool Footer 3	Query	Footer		Apr 13, 2021 11:09 AM	catavolt
ProjTool Footer 4	ProjTool Footer 4	Query	Footer		Apr 13, 2021 11:11 AM	catavolt

Figure 13: Maintainable Query Dashboard Sections view Maintainable Section Type highlighted

Header(s) are shown first, followed by the Data section, followed by Footer(s). All sections are laid out and scrolled together horizontally as if they are a single section. In order to ensure that each section lines up and looks synchronous with the others, Queries in each section should have the same number of columns, column spacers should appear in the same locations, etc.

Heading Groups



Home / Dashboards / Dashboard: ProjTool2 / Projects

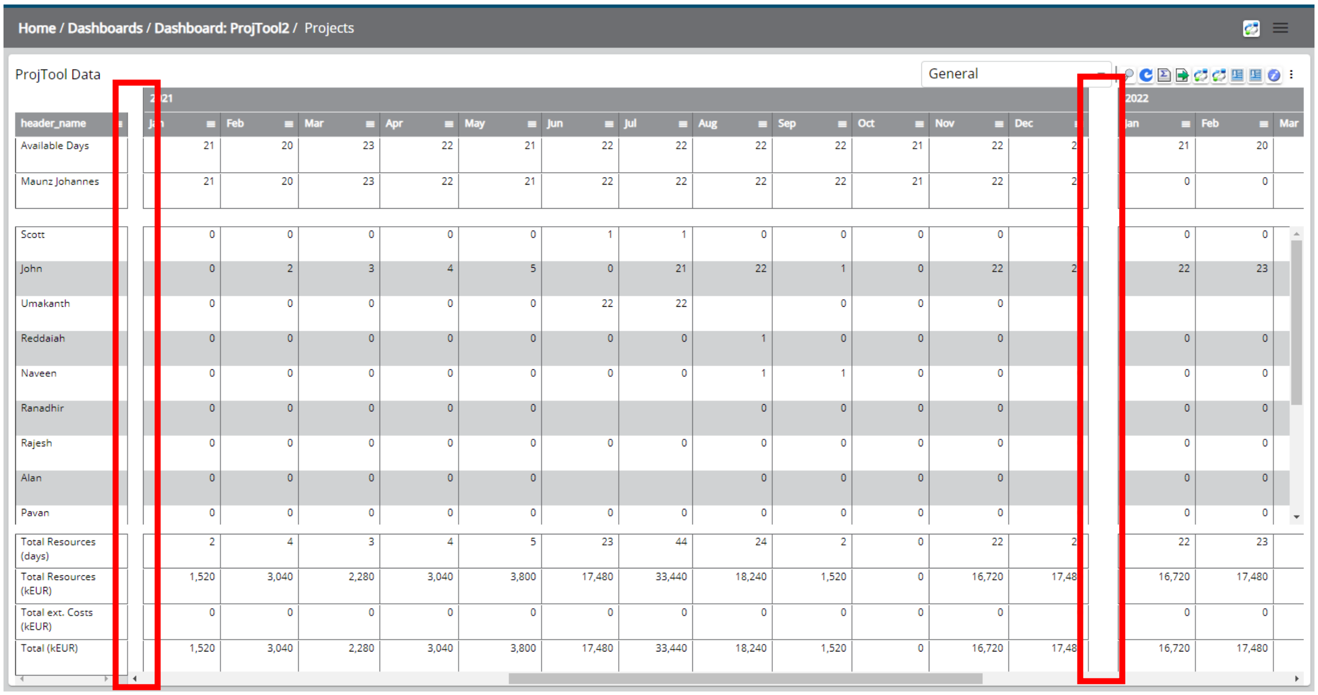
ProjTool Data

	2021												2022		
header_name	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar
Available Days	21	20	23	22	21	22	22	22	22	21	22	23	21	20	
Maunz Johannes	21	20	23	22	21	22	22	22	22	21	22	23	0	0	
Scott	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
John	0	2	3	4	5	0	21	22	1	0	22	21	22	23	
Umakanth	0	0	0	0	0	22	22		0	0	0	1			
Reddaiah	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
Naveen	0	0	0	0	0	0	0	1	1	0	0	0	0	0	
Ranadhir	0	0	0	0	0			0	0	0	0	0	0	0	
Rajesh	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Alan	0	0	0	0	0			0	0	0	0	1	0	0	
Pavan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total Resources (days)	2	4	3	4	5	23	44	24	2	0	22	23	22	23	
Total Resources (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	
Total ext. Costs (kEUR)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	

Figure 14: Maintainable Query with Heading Groups highlighted

Heading Groups are specified on Query View Properties. In the example above, all properties in the highlighted area have "2021" as a Heading Group.

Spacer Columns



Home / Dashboards / Dashboard: ProjTool2 / Projects

ProjTool Data

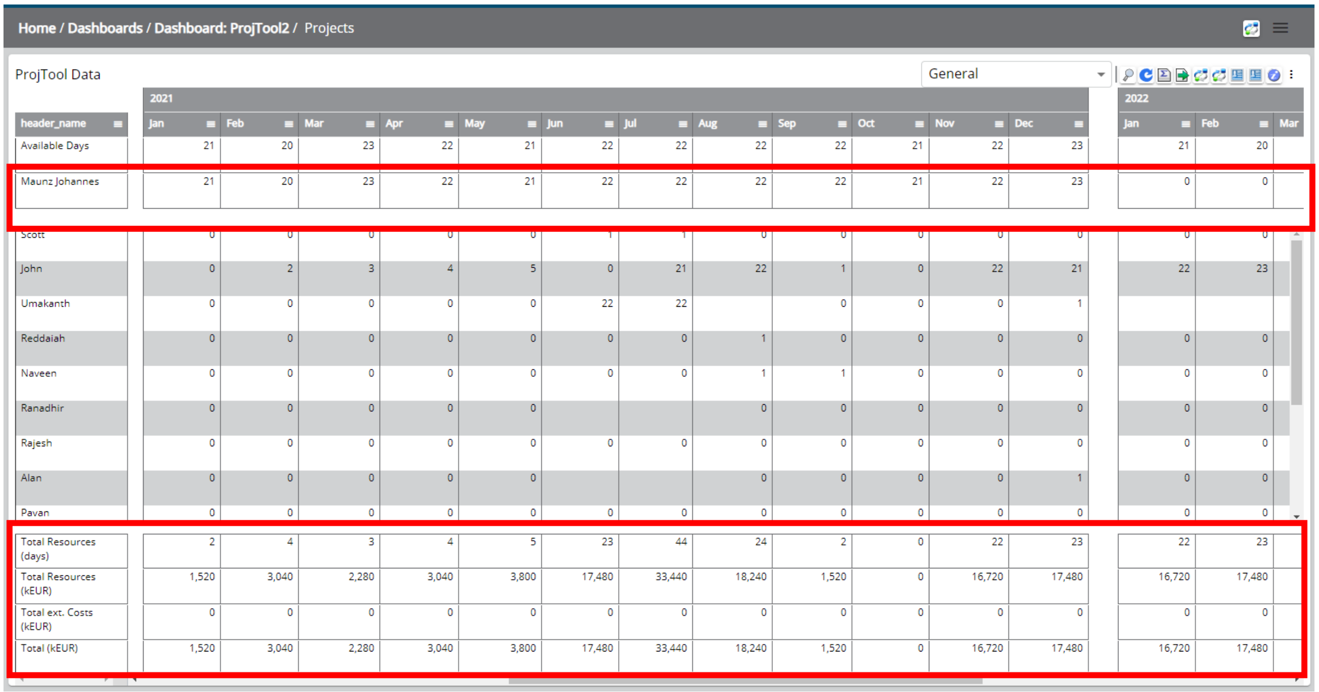
General

		2021													2022		
header_name		Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec		Jan	Feb	Mar
Available Days		21	20	23	22	21	22	22	22	22	21	22	2		21	20	
Maunz Johannes		21	20	23	22	21	22	22	22	22	21	22	2		0	0	
Scott		0	0	0	0	0	1	1	0	0	0	0			0	0	
John		0	2	3	4	5	0	21	22	1	0	22	2		22	23	
Umakanth		0	0	0	0	0	22	22		0	0	0					
Reddaiah		0	0	0	0	0	0	0	1	0	0	0			0	0	
Naveen		0	0	0	0	0	0	0	1	1	0	0			0	0	
Ranadhir		0	0	0	0	0			0	0	0	0			0	0	
Rajesh		0	0	0	0	0	0	0	0	0	0	0			0	0	
Alan		0	0	0	0	0			0	0	0	0			0	0	
Pavan		0	0	0	0	0	0	0	0	0	0	0			0	0	
Total Resources (days)		2	4	3	4	5	23	44	24	2	0	22	2		22	23	
Total Resources (kEUR)		1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,48		16,720	17,480	
Total ext. Costs (kEUR)		0	0	0	0	0	0	0	0	0	0	0			0	0	
Total (kEUR)		1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,48		16,720	17,480	

Figure 15: Maintainable Query with Spacer Columns highlighted

Column Spacers are specified on a Query using the "Add Maintainable Query Spacer Column" menu option. They are rendered as a blank column with no Heading.

Spacer Lines / Suppress Heading



Home / Dashboards / Dashboard: ProjTool2 / Projects

ProjTool Data

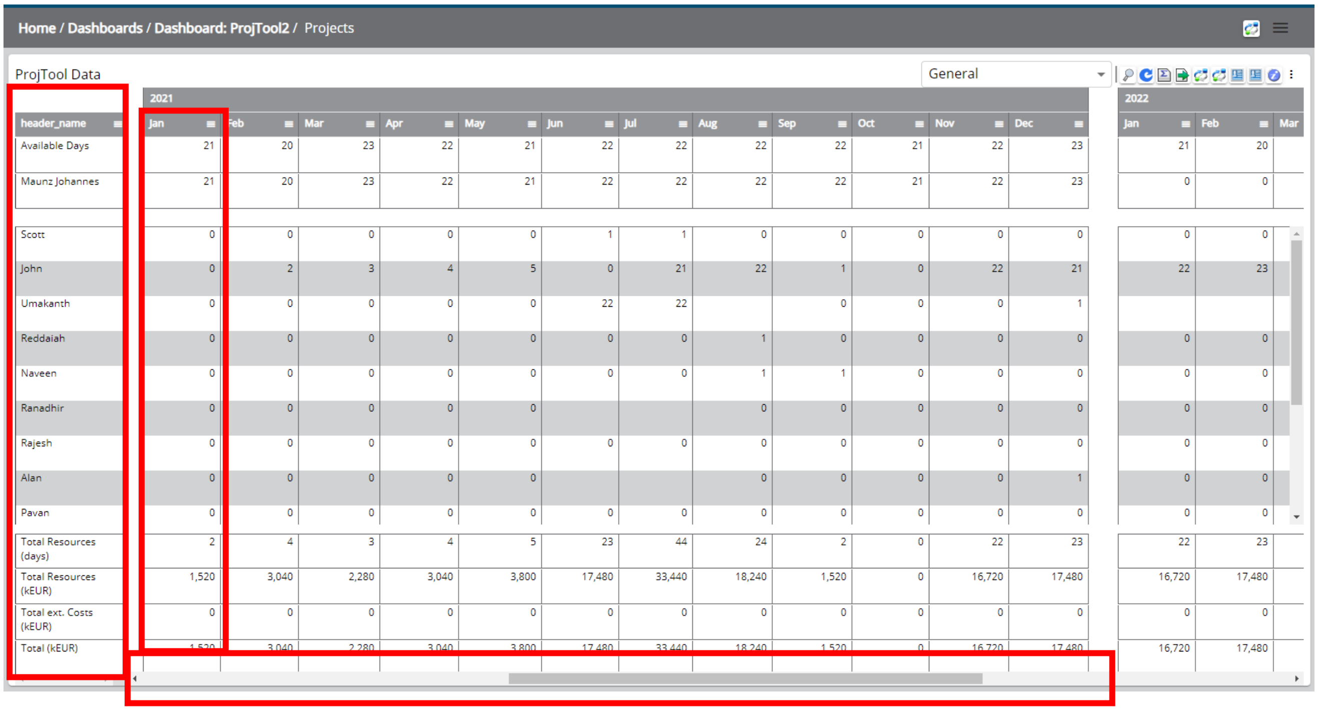
General

	2021												2022		
header_name	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar
Available Days	21	20	23	22	21	22	22	22	22	21	22	23	21	20	
Maunz Johannes	21	20	23	22	21	22	22	22	22	21	22	23	0	0	
Scott	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
John	0	2	3	4	5	0	21	22	1	0	22	21	22	23	
Umakanth	0	0	0	0	0	22	22		0	0	0	1			
Reddaiah	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
Naveen	0	0	0	0	0	0	0	1	1	0	0	0	0	0	
Ranadhir	0	0	0	0	0			0	0	0	0	0	0	0	
Rajesh	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Alan	0	0	0	0	0			0	0	0	0	1	0	0	
Pavan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total Resources (days)	2	4	3	4	5	23	44	24	2	0	22	23	22	23	
Total Resources (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	
Total ext. Costs (kEUR)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	

Figure 16: Maintainable Query with Spacer Lines / Suppress Heading highlighted

Spacer Lines specify how many lines to separate the current section from the following section. In the top highlight, you can see that Spacer Lines is set to 1. In the bottom highlight, there are 4 Footer sections all with Spacer Lines set to 0. The Footer sections also have Suppress Heading = Yes. The combination of these 2 values make the 4 Footer sections appear to lay out as a single table.

Initial Display Options / Maintainable



Home / Dashboards / Dashboard: ProjTool2 / Projects

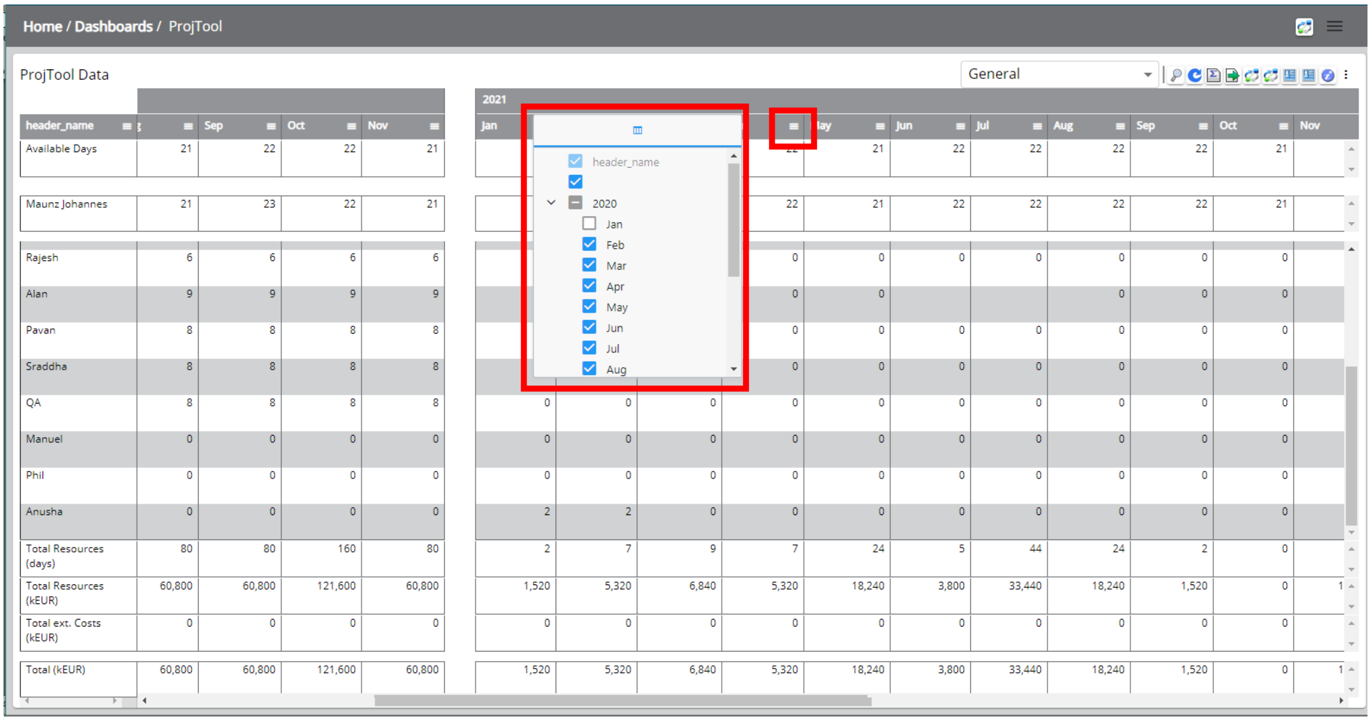
ProjTool Data

General

	2021												2022		
header_name	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	Jan	Feb	Mar
Available Days	21	20	23	22	21	22	22	22	22	21	22	23	21	20	
Maunz Johannes	21	20	23	22	21	22	22	22	22	21	22	23	0	0	
Scott	0	0	0	0	0	1	1	0	0	0	0	0	0	0	
John	0	2	3	4	5	0	21	22	1	0	22	21	22	23	
Umakanth	0	0	0	0	0	22	22		0	0	0	1			
Reddaiah	0	0	0	0	0	0	0	1	0	0	0	0	0	0	
Naveen	0	0	0	0	0	0	0	1	1	0	0	0	0	0	
Ranadhir	0	0	0	0	0			0	0	0	0	0	0	0	
Rajesh	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Alan	0	0	0	0	0			0	0	0	0	1	0	0	
Pavan	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total Resources (days)	2	4	3	4	5	23	44	24	2	0	22	23	22	23	
Total Resources (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	
Total ext. Costs (kEUR)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total (kEUR)	1,520	3,040	2,280	3,040	3,800	17,480	33,440	18,240	1,520	0	16,720	17,480	16,720	17,480	

Figure 17: Maintainable Query with Pinned / First Visible Columns highlighted

Initial Display Options are specified on Query View Properties. In the example above, the Resource Name column has an Initial Display option of Pinned, so it is not affected by the horizontal scrollbar. Likewise, Jan 2021 has an Initial Display option of First Visible. As you can see, even though 2020, 2021, and 2022 properties are all defined on the Query, the horizontal scrollbar is initially positioned so that Jan 2021 is the first non-pinned column to be displayed.



ProjTool Data

					2021										
header_name	g	Sep	Oct	Nov	Jan				lay	Jun	Jul	Aug	Sep	Oct	Nov
Available Days	21	22	22	21				22	21	22	22	22	22	21	
Maunz Johannes	21	23	22	21				22	21	22	22	22	22	21	
Rajesh	6	6	6	6				0	0	0	0	0	0	0	
Alan	9	9	9	9				0	0			0	0	0	
Pavan	8	8	8	8				0	0	0	0	0	0	0	
Sraddha	8	8	8	8				0	0	0	0	0	0	0	
QA	8	8	8	8	0	0	0	0	0	0	0	0	0	0	
Manuel	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Phil	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Anusha	0	0	0	0	2	2	0	0	0	0	0	0	0	0	
Total Resources (days)	80	80	160	80	2	7	9	7	24	5	44	24	2	0	
Total Resources (kEUR)	60,800	60,800	121,600	60,800	1,520	5,320	6,840	5,320	18,240	3,800	33,440	18,240	1,520	0	1
Total ext. Costs (kEUR)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
Total (kEUR)	60,800	60,800	121,600	60,800	1,520	5,320	6,840	5,320	18,240	3,800	33,440	18,240	1,520	0	1

To manipulate Hidden columns, click on the icon on the right of any column. This will bring up a menu of all columns. Hidden columns are unchecked, while columns currently visible are checked. You can select/unselect any columns in this menu to dynamically show/hide them on the underlying list.

Maintainable values (which columns can be edited) are also specified on Query View Properties.

Note that Initial Display Options and Maintainable values for each column are pulled from the **Data** section's Query. Any values set on Header or Footer sections will be ignored. Also note that Header and Footer sections are (by definition) read-only sections. No cells in these sections can be maintained regardless of the Maintainable value.

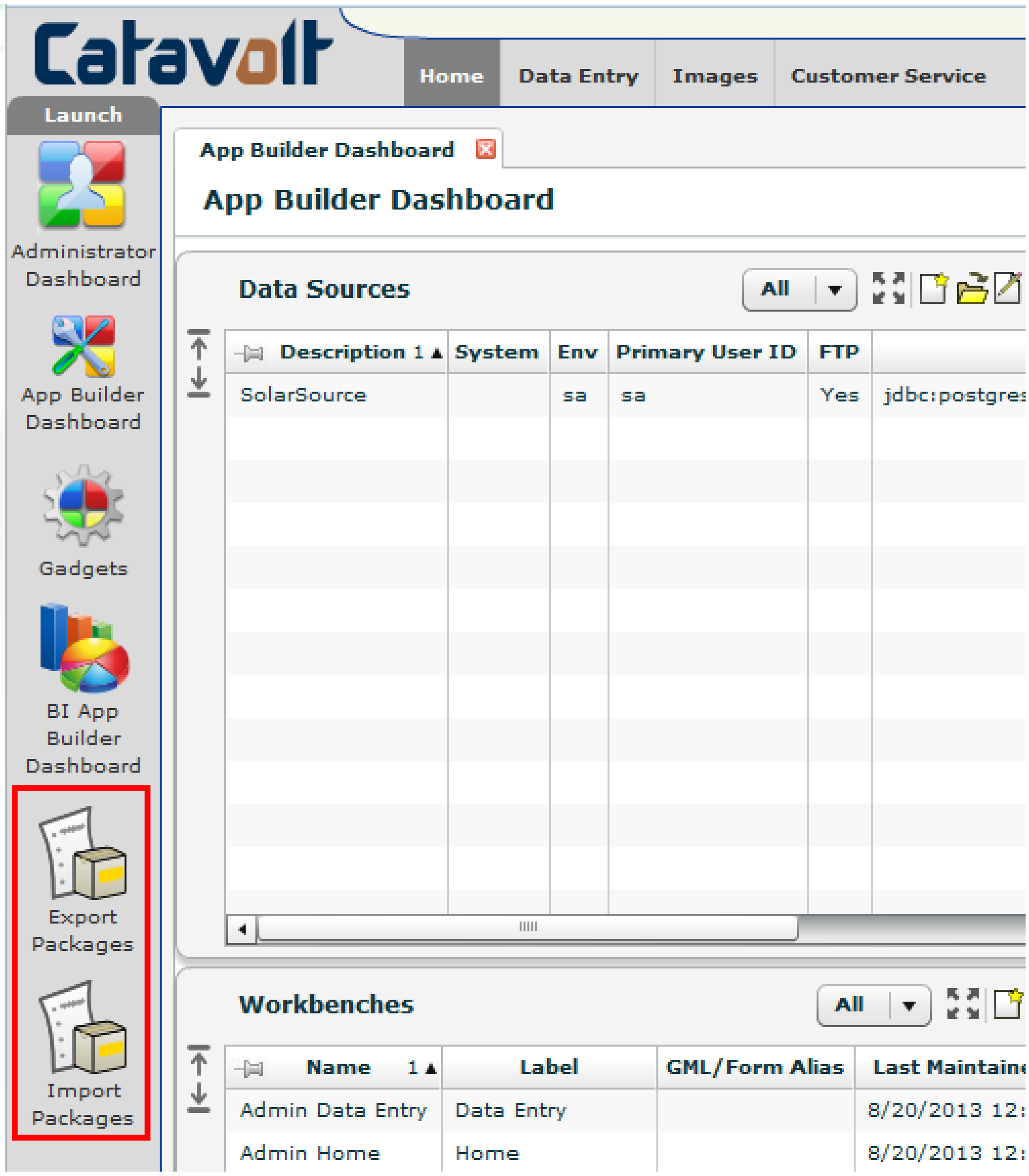
Chapter 15: Import/Export

Chapter Summary

Xalt | Mobility has added the ability to export objects from one environment and import them into the same or another environment. This is primarily used for customers who maintain multiple tenants (for example, development and production environments) and need a way to move Data Objects, Data Sources, Workbenches, etc. from one tenant to another. You also have the ability to import objects into the same environment, which allows you to perform functions that are not normally allowed (e.g. copying a Data Object from one Data Source to another).

Accessing Import/Export

All import/export functions are accessed through two launchers on the Admin Home workbench – Export Packages and Import Packages.



App Builder Dashboard

Data Sources

Description 1 ▲	System	Env	Primary User ID	FTP	
SolarSource		sa	sa	Yes	jdbc:postgre

Workbenches

Name 1 ▲	Label	GML/Form Alias	Last Maintaine
Admin Data Entry	Data Entry		8/20/2013 12:
Admin Home	Home		8/20/2013 12:

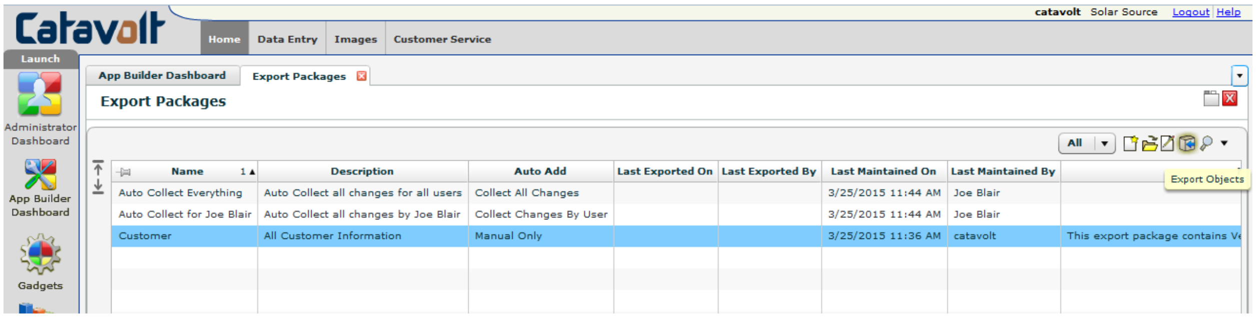
Figure 1: Export/Import Package launchers

Export

When performing an Export, you must first create an Export Package. An Export Package defines the contents of the export. Once complete, you can then call the Export Objects action, which will package up the data and create an export ZIP file that will be used during the Import process. The following objects can be included in an Export Package:

- Data Object
- Data Source
- Connector gateway
- Workbench
- Gadget
- Dashboard
- Default Property

In addition, any Custom Forms, Images, and Security Roles that are referenced by objects in the Export Package will be automatically included.



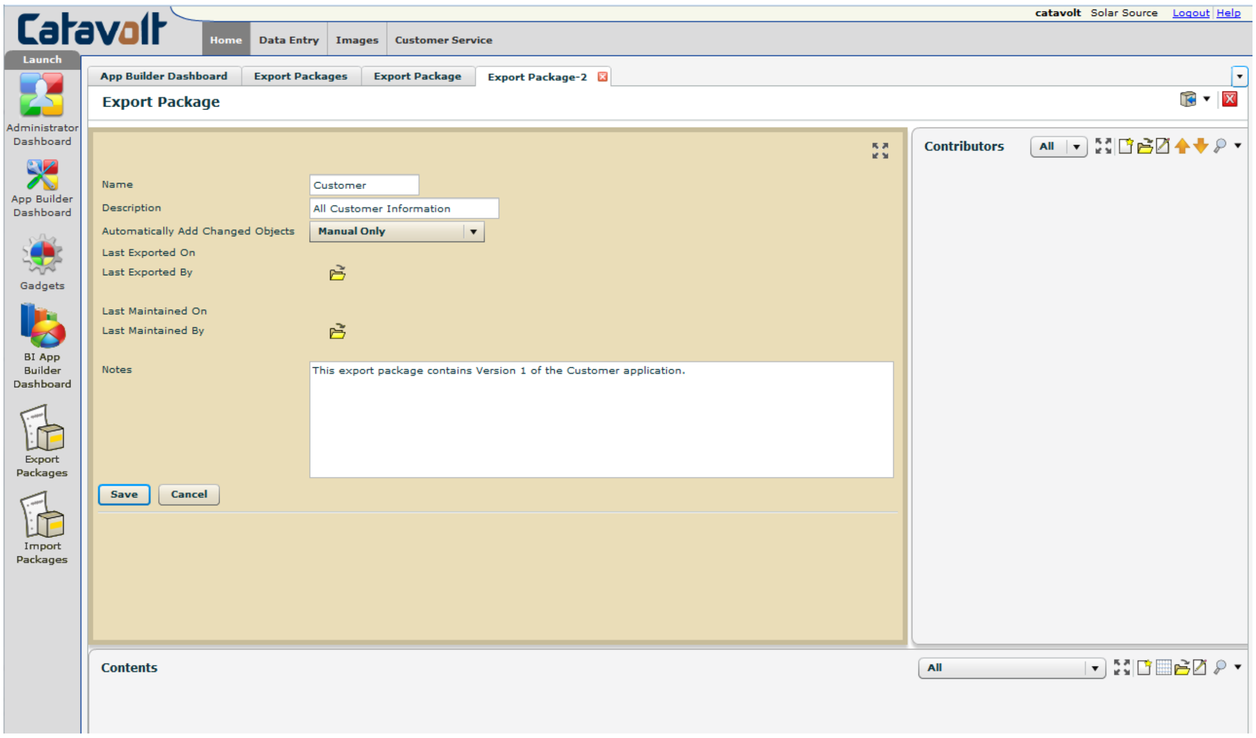
Export Packages

Name	Description	Auto Add	Last Exported On	Last Exported By	Last Maintained On	Last Maintained By	
Auto Collect Everything	Auto Collect all changes for all users	Collect All Changes			3/25/2015 11:44 AM	Joe Blair	
Auto Collect for Joe Blair	Auto Collect all changes by Joe Blair	Collect Changes By User			3/25/2015 11:44 AM	Joe Blair	
Customer	All Customer Information	Manual Only			3/25/2015 11:36 AM	catavolt	This export package contains Ve

Figure 2: Export Packages list view

Creating Export Packages

To create an Export Package, choose the New menu option.



Name: Customer
Description: All Customer Information
Automatically Add Changed Objects: Manual Only
Last Exported On
Last Exported By
Last Maintained On
Last Maintained By
Notes: This export package contains Version 1 of the Customer application.

Figure 3: Export Package create details view

Name specifies the name of the Export Package. This value is used to uniquely identify the Export Package.

Description specifies a brief description of the Export Package.

Automatically Add Changed Objects specifies whether you want to automatically add objects to the Export package as they are modified in Xalt. This allows you to have one or more Export Packages automatically "collect" development changes as they are being made. The following options are available:

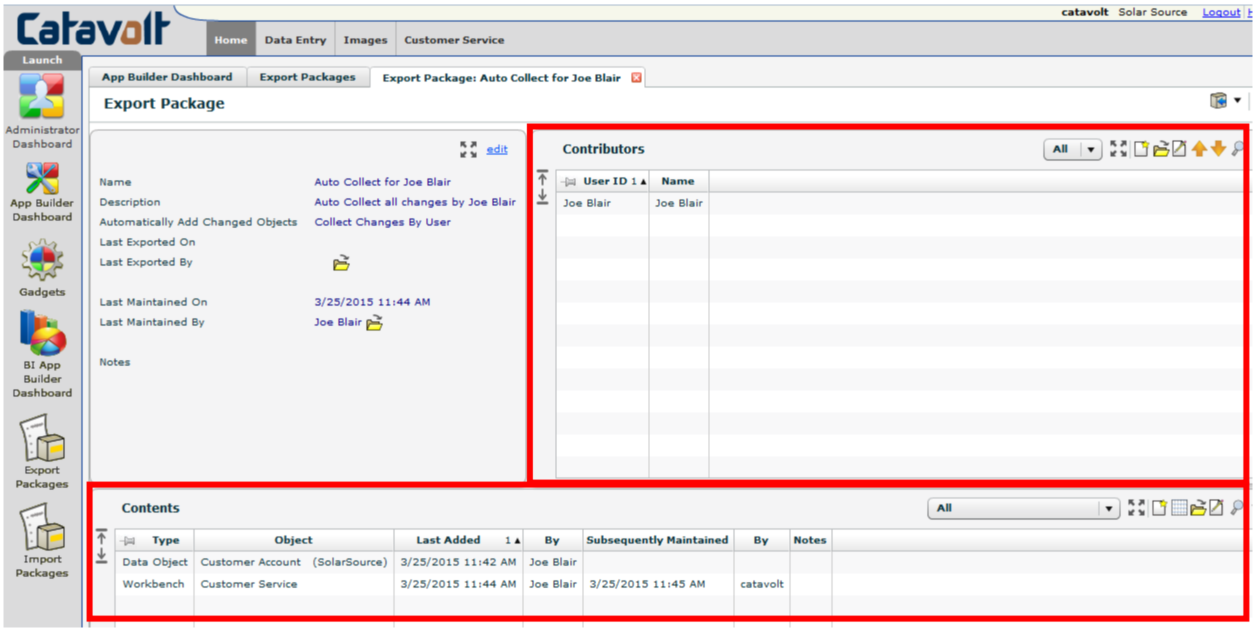
- Manual Only – Objects are manually added to the Export Package
- Collect All Changes – Objects are automatically added to the Export Package when they are changed by any user.
- Collect Changes by User – Objects are automatically added to the Export Package when they are changed by any user in the Contributors list.

Note that Connector gateways and Data Sources will not be automatically added to an Export Package regardless of the setting chosen. These must be added Manually if you wish to include them in the Export Package.

Notes allows you to specify a more detailed description and/or other things to be noted with this Export Package.

Export Package Components

An Export Package is made up of 2 components: Contents (the objects that are to be exported), and Contributors (a list of User Profiles whose development changes will be automatically collected in the Export Package).



Export Package

Name	Auto Collect for Joe Blair
Description	Auto Collect all changes by Joe Blair
Automatically Add Changed Objects	Collect Changes By User
Last Exported On	
Last Exported By	
Last Maintained On	3/25/2015 11:44 AM
Last Maintained By	Joe Blair
Notes	

Contributors

User ID 1 ▲	Name
Joe Blair	Joe Blair

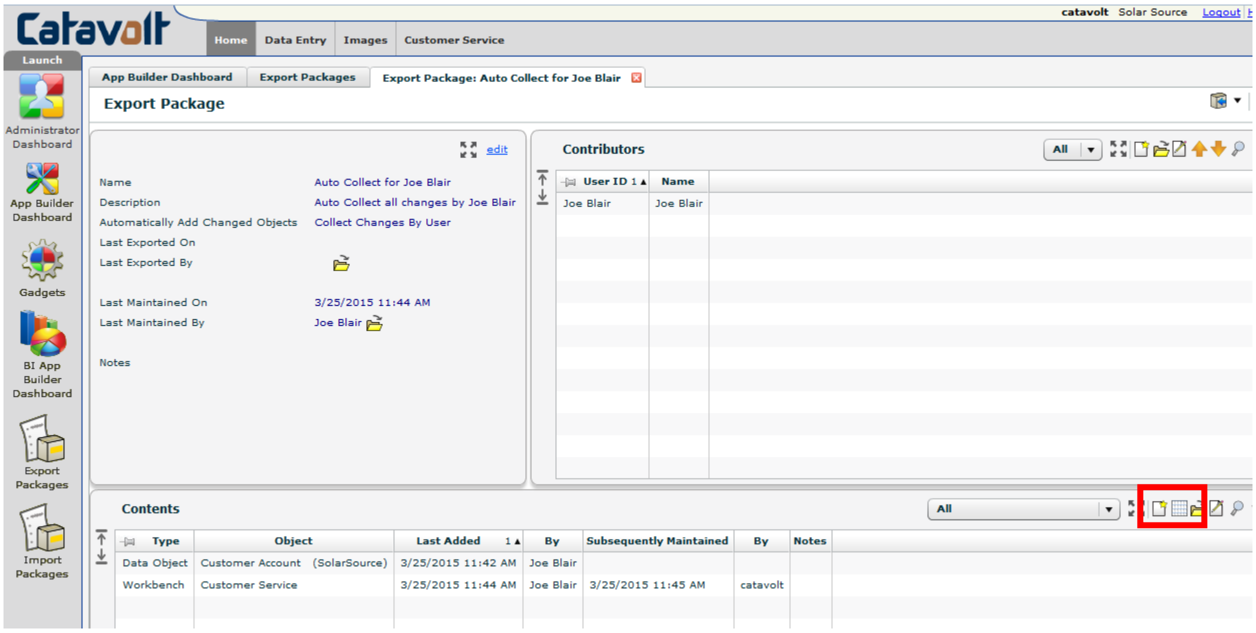
Contents

Type	Object	Last Added 1 ▲	By	Subsequently Maintained	By	Notes
Data Object	Customer Account (SolarSource)	3/25/2015 11:42 AM	Joe Blair			
Workbench	Customer Service	3/25/2015 11:44 AM	Joe Blair	3/25/2015 11:45 AM	catavolt	

Figure 4: Export Package details view with the Contributors and Contents sections highlighted

Contents

The Contents section allows you to specify one or more objects that are to be included in the Export Package. Objects will be added to the Contents section if the Automatically Add Changed Objects property is set to **Collect All Changes** or **Collect Changed By User**. You can also manually add objects regardless of the setting by choosing the New and Add Content By Date actions.



Export Package

Name	Auto Collect for Joe Blair
Description	Auto Collect all changes by Joe Blair
Automatically Add Changed Objects	Collect Changes By User
Last Exported On	
Last Exported By	
Last Maintained On	3/25/2015 11:44 AM
Last Maintained By	Joe Blair
Notes	

Contributors

User ID 1 ▲	Name
Joe Blair	Joe Blair

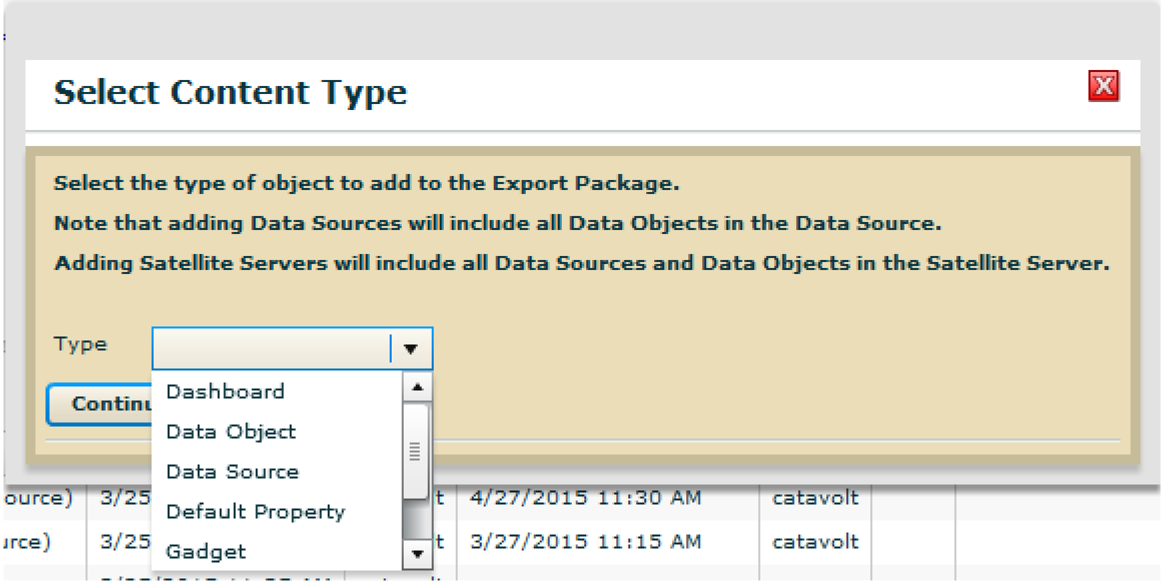
Contents

Type	Object	Last Added 1 ▲	By	Subsequently Maintained	By	Notes
Data Object	Customer Account (SolarSource)	3/25/2015 11:42 AM	Joe Blair			
Workbench	Customer Service	3/25/2015 11:44 AM	Joe Blair	3/25/2015 11:45 AM	catavolt	

Figure 5: An Export Package details view with the New and Add Content By Date actions highlighted

Adding Content

To add objects to an Export Package, choose the New option on the Content list. You will be presented with the following screen:



Select Content Type

Select the type of object to add to the Export Package.

Note that adding Data Sources will include all Data Objects in the Data Source.

Adding Satellite Servers will include all Data Sources and Data Objects in the Satellite Server.

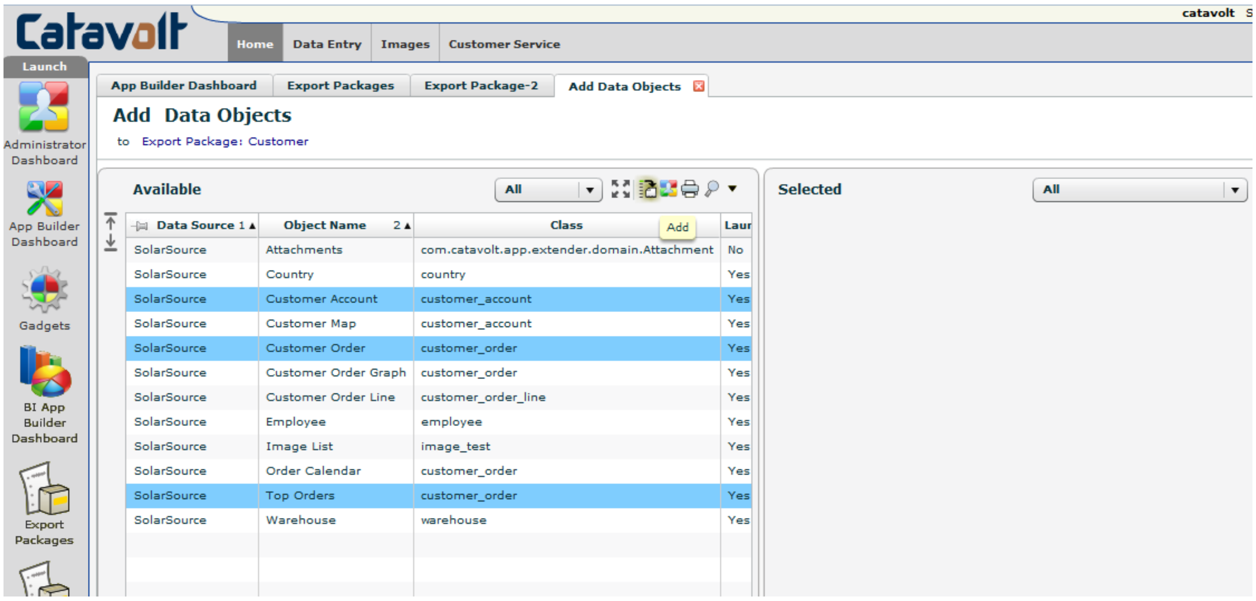
Figure 6: The Add Export Package Content prompt

You must first select the Type of object to add to the Export Package. The following objects can be included in an Export Package:

- Dashboard
- Data Object
- Data Source
- Connector gateway
- Workbench
- Gadget
- Default Property

Note that adding a Data Source will **automatically** include all Data Objects in that Data Source. Similarly, adding a Connector gateway will **automatically** include all Data Sources and Data Objects in that Connector gateway.

Once you select the type of object to add, you will be presented with a screen like the following:



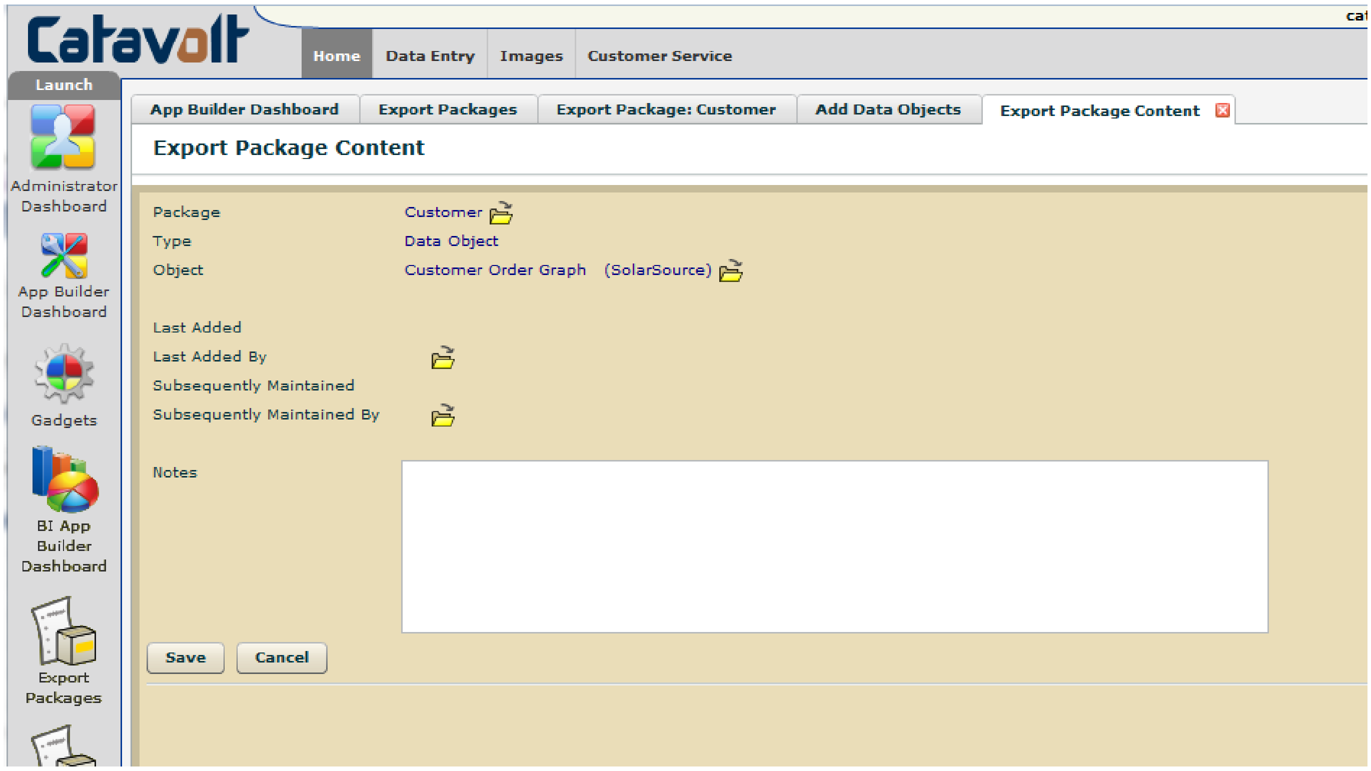
Add Data Objects

to Export Package: Customer

Data Source 1▲	Object Name 2▲	Class	Laur
SolarSource	Attachments	com.catavolt.app.extender.domain.Attachment	No
SolarSource	Country	country	Yes
SolarSource	Customer Account	customer_account	Yes
SolarSource	Customer Map	customer_account	Yes
SolarSource	Customer Order	customer_order	Yes
SolarSource	Customer Order Graph	customer_order	Yes
SolarSource	Customer Order Line	customer_order_line	Yes
SolarSource	Employee	employee	Yes
SolarSource	Image List	image_test	Yes
SolarSource	Order Calendar	customer_order	Yes
SolarSource	Top Orders	customer_order	Yes
SolarSource	Warehouse	warehouse	Yes

Figure 7: The Add Data Object to Export Package menu option

You can select one of more objects of that type and add them to the Export Package. If you select a single object, you will see the following screen where you can add Notes to the content. Any notes you enter here will be carried over to the Import package.



Export Package Content

Package Customer

Type Data Object

Object Customer Order Graph (SolarSource)

Last Added

Last Added By

Subsequently Maintained

Subsequently Maintained By

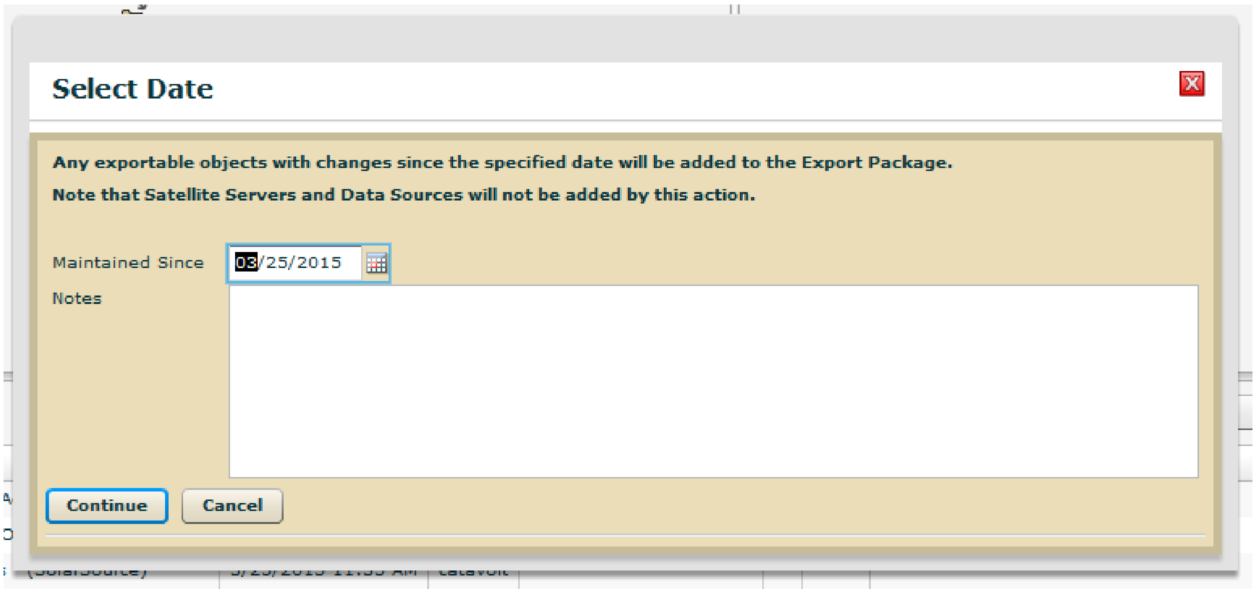
Notes

Figure 8: The Add Content view

You can add objects of different types to the same Export Package. For example, your Export Package can contain 5 Data Objects, 2 Workbenches, and 3 Default Properties if desired.

Adding Content by Date

To add objects by date to an Export Package, choose the New option on the Content list. You will be presented with the following screen:



Select Date

Any exportable objects with changes since the specified date will be added to the Export Package.

Note that Satellite Servers and Data Sources will not be added by this action.

Maintained Since 03/25/2015

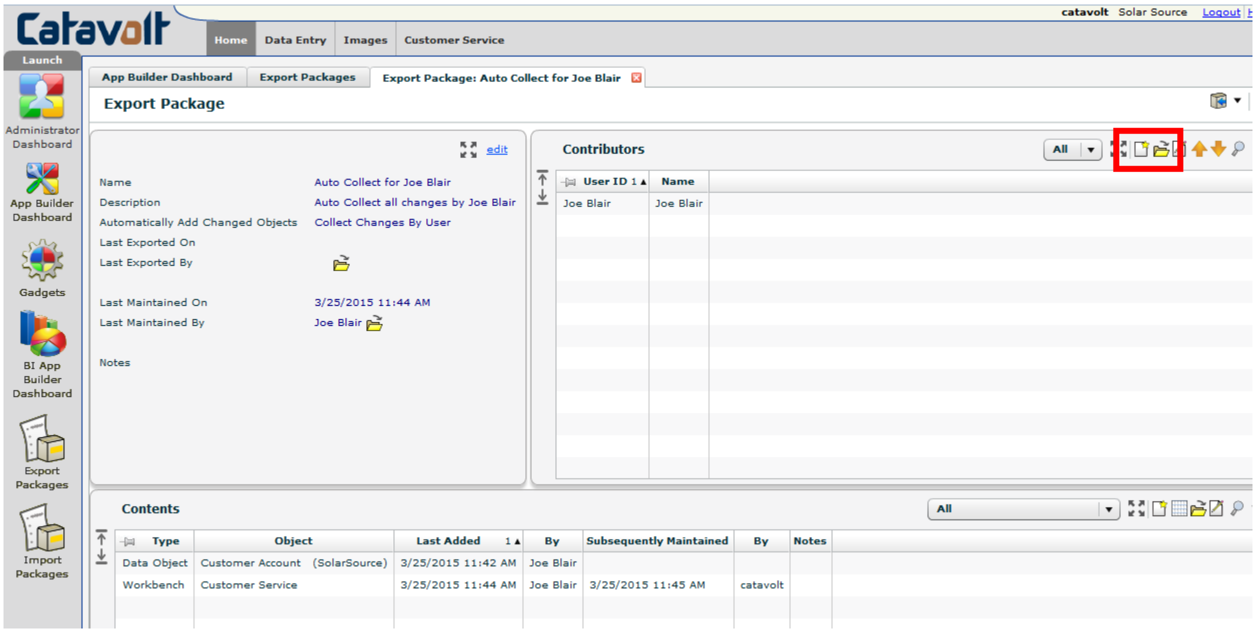
Notes

Figure 9: The add Content By Date view

You can specify a Maintained Since date. Any object (other than Connector gateways and Data Sources) that has been changed on or after this date will be added to the Export package. You can optionally add Notes to the objects as they are added to the Export Package.

Contributors

The Contributors section allows you to specify one or more users whose Xalt changes will be automatically be collected in the Export Package. You are able to specify users if the Automatically Add Changed Objects property is set to **Collect Changed By User**. This function is useful in cases where you have multiple projects going on simultaneously in the same environment that need to be exported separately.



Export Package

Name	Auto Collect for Joe Blair
Description	Auto Collect all changes by Joe Blair
Automatically Add Changed Objects	Collect Changes By User
Last Exported On	
Last Exported By	
Last Maintained On	3/25/2015 11:44 AM
Last Maintained By	Joe Blair
Notes	

Contributors

User ID 1 ▲	Name
Joe Blair	Joe Blair

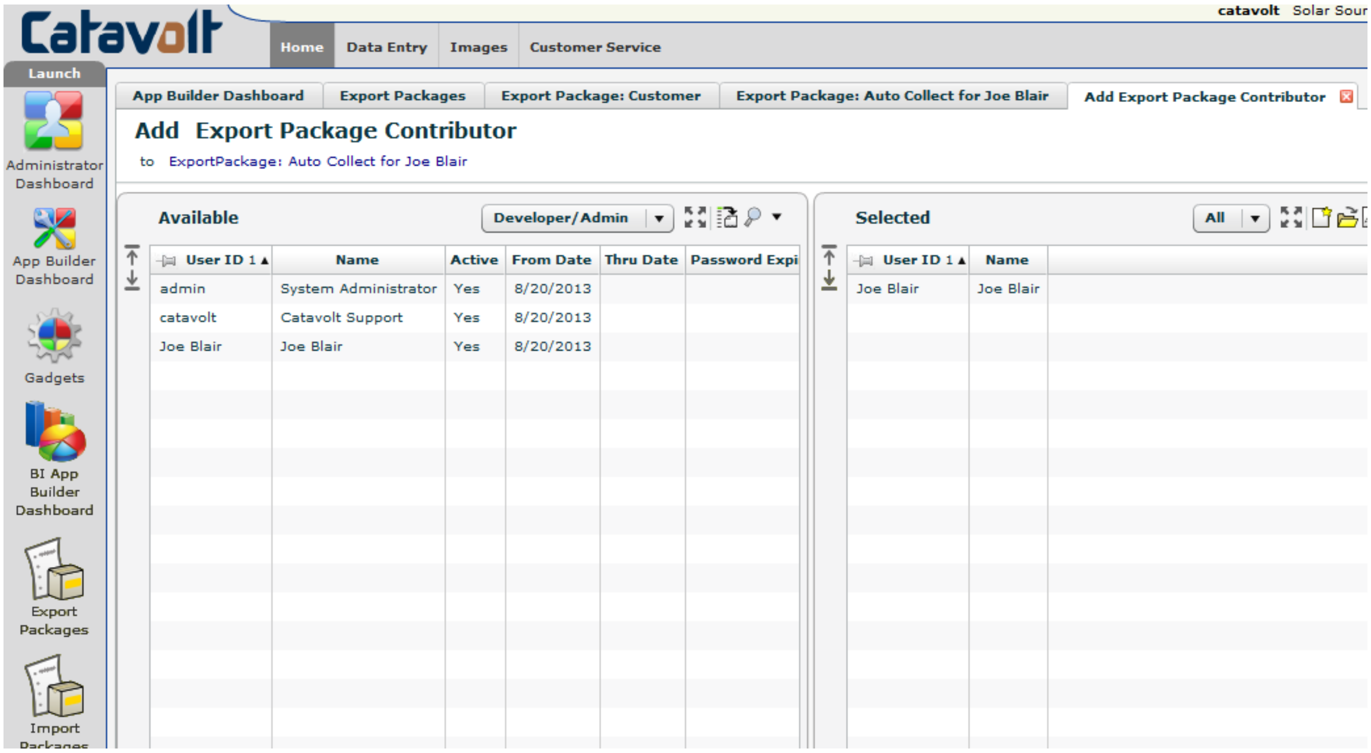
Contents

Type	Object	Last Added 1 ▲	By	Subsequently Maintained	By	Notes
Data Object	Customer Account (SolarSource)	3/25/2015 11:42 AM	Joe Blair			
Workbench	Customer Service	3/25/2015 11:44 AM	Joe Blair	3/25/2015 11:45 AM	catavolt	

Figure 10: An Export Package details view with the New Contributor action highlighted

Adding Contributors

To add a Contributor to an Export Package, choose the New option on the Contributors list. You will be presented with the following screen:



Add Export Package Contributor

to ExportPackage: Auto Collect for Joe Blair

Available

User ID 1 ▲	Name	Active	From Date	Thru Date	Password Expi
admin	System Administrator	Yes	8/20/2013		
catavolt	Catavolt Support	Yes	8/20/2013		
Joe Blair	Joe Blair	Yes	8/20/2013		

Selected

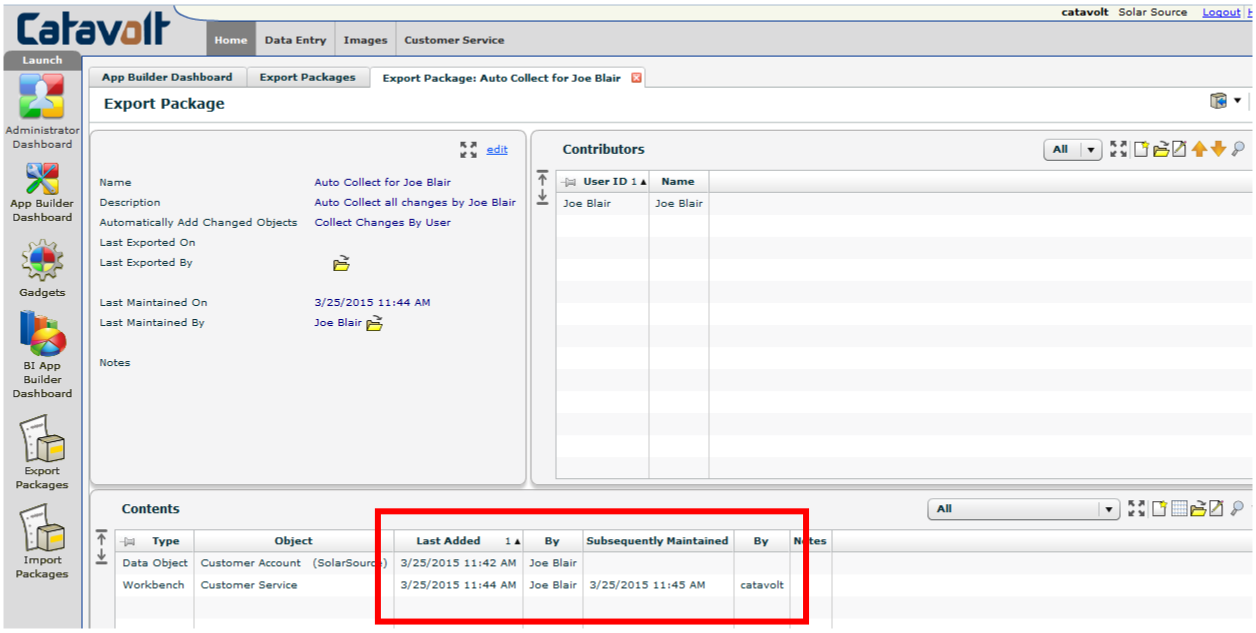
User ID 1 ▲	Name
Joe Blair	Joe Blair

Figure 11: The Add Export Package Contributor prompt

You can select one or more User Profiles to be contributors. Any Xalt changes these users make will automatically be added to the Export Package.

Last Added/Subsequently Maintained

If the Export Package has Automatically Add Changed Objects set to Collect All Changes, then each time the object is changed, the Last Added and Last Added By properties are updated to reflect the last change. If the Export Package has Automatically Add Changed Objects set to Collect Changed By User, then each time the object is changed by a user in the Contributor list, the Last Added and Last Added By properties are updated to reflect the last change. If the object is changed by a user not in the Contributor list, the Subsequently Maintained and Subsequently Maintained By properties are updated to reflect the last change. This gives you a convenient way to know that someone else has made changes to an object that you are planning to export.



Export Package

Name	Auto Collect for Joe Blair
Description	Auto Collect all changes by Joe Blair
Automatically Add Changed Objects	Collect Changes By User
Last Exported On	
Last Exported By	
Last Maintained On	3/25/2015 11:44 AM
Last Maintained By	Joe Blair
Notes	

Contributors

User ID 1 ▲	Name
Joe Blair	Joe Blair

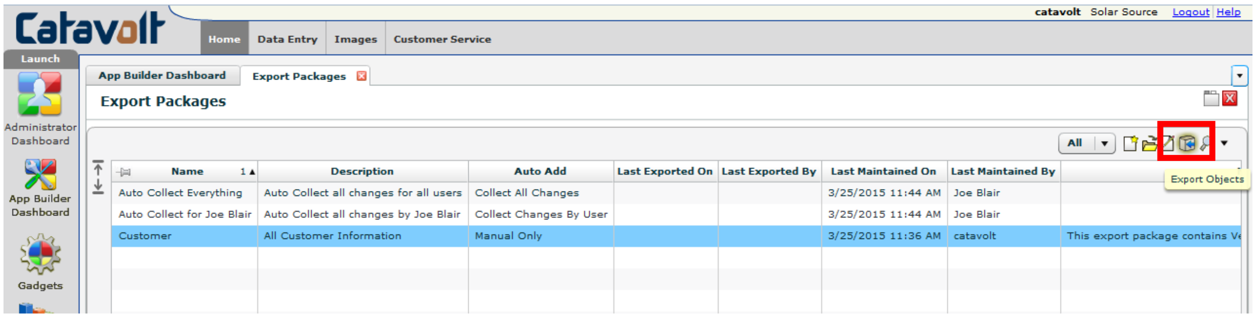
Contents

Type	Object	Last Added 1 ▲	By	Subsequently Maintained	By	Notes
Data Object	Customer Account (SolarSource)	3/25/2015 11:42 AM	Joe Blair			
Workbench	Customer Service	3/25/2015 11:44 AM	Joe Blair	3/25/2015 11:45 AM	catavolt	

Figure 12: The Export Package detail view with the Last Added/Subsequently Maintained fields highlighted

Performing an Export

When your Export Package is ready, choose the Export Objects menu action.

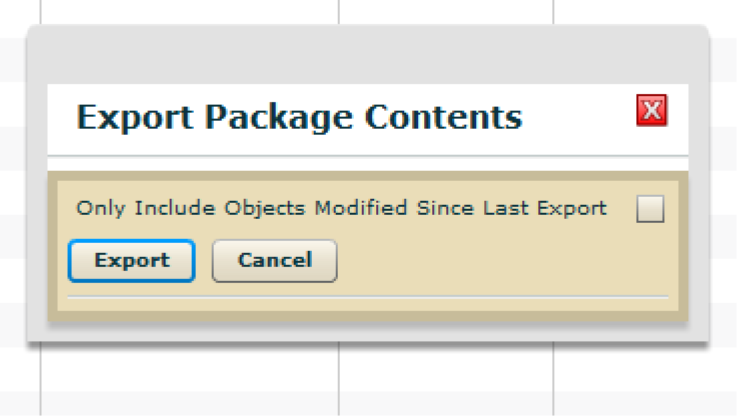


Export Packages

Name 1 ▲	Description	Auto Add	Last Exported On	Last Exported By	Last Maintained On	Last Maintained By	
Auto Collect Everything	Auto Collect all changes for all users	Collect All Changes			3/25/2015 11:44 AM	Joe Blair	
Auto Collect for Joe Blair	Auto Collect all changes by Joe Blair	Collect Changes By User			3/25/2015 11:44 AM	Joe Blair	
Customer	All Customer Information	Manual Only			3/25/2015 11:36 AM	catavolt	This export package contains Ve

Figure 13: The Export Package list view with the Export Objects menu action highlighted

You will be presented with the following window:

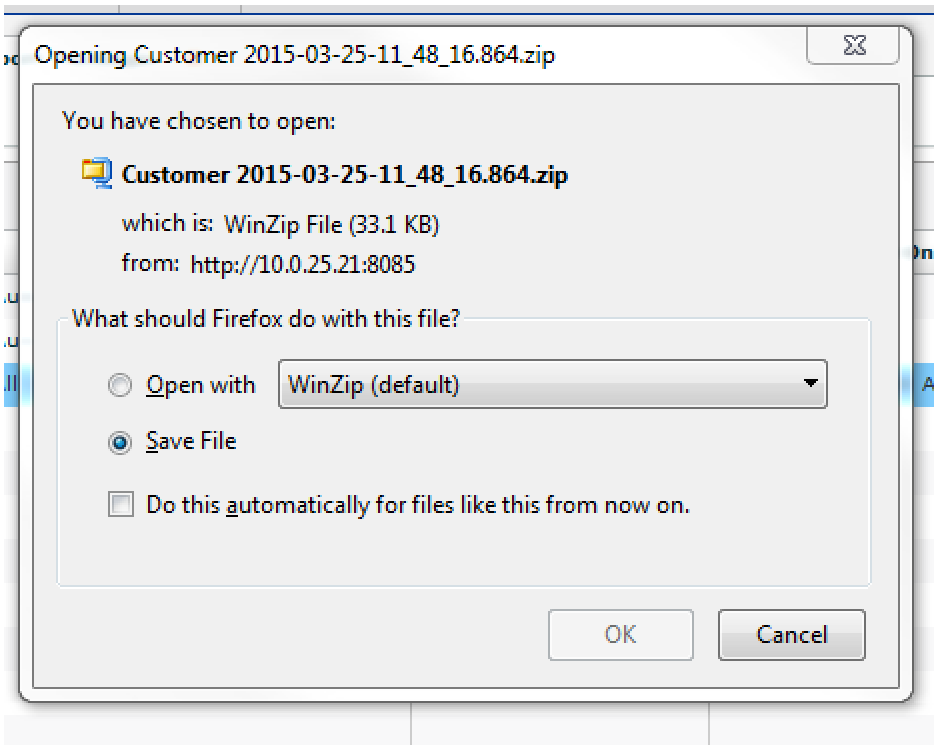


Export Package Contents

Only Include Objects Modified Since Last Export □

Figure 14: The Export Package Contents view

After the initial export, you can choose to re-export everything or select Only Include Objects Modified Since Last Export to only export objects that have been changed since the last export. The export process will collect all the objects and put them into a ZIP file, then prompt you to save the file to your local machine. While processing your objects, Xalt will also pull any Custom Forms, Images, and Security Profiles that are being referenced by your objects and includes them in the ZIP file as well.



Opening Customer 2015-03-25-11_48_16.864.zip

You have chosen to open:

Customer 2015-03-25-11_48_16.864.zip

which is: WinZip File (33.1 KB)

from: http://10.0.25.21:8085

What should Firefox do with this file?

○ Open with WinZip (default)

● Save File

□ Do this automatically for files like this from now on.

Figure 15: The Save As view for the produced ZIP file

Export Tips and Best Practices

When dealing with multiple environments, you should view Export as a one-way process. All work should be done in one environment (development), then exported over to the other (production). In theory, working in both environments and moving changes back and forth will work, but in practice it will make things more complicated and you will not always get the behavior you are expecting.

You may want to consider setting up Data Sources as "applications". You can have multiple Data Sources over the same back-end system. Exporting an entire Data Source will be easier to import vs exporting multiple individual Data Objects.

If you have an Export Package that is collecting changes automatically, you can turn off the behavior by setting Automatically Add Changed Objects to **Manual Only**. This is a useful way to "archive" an Export Package once you have exported it.

Any Notes you add to an Export Package or its Contents will be automatically carried over to the Import Package. This is handy if you want to make some notes during Export that you can refer to during Import.

If a single developer is working on two simultaneous projects to be exported separately, you can create two User Profiles for the developer, one for each project. You can then create two Export Packages, one collecting changes from each User Profile. This will help separate out the projects without having to manually add objects to your Export Packages.

An Export Package can be reused multiple times to export the same set of objects over and over again.

BI Objects (Data Sets, Dashboards, etc.) are currently not available to be exported.

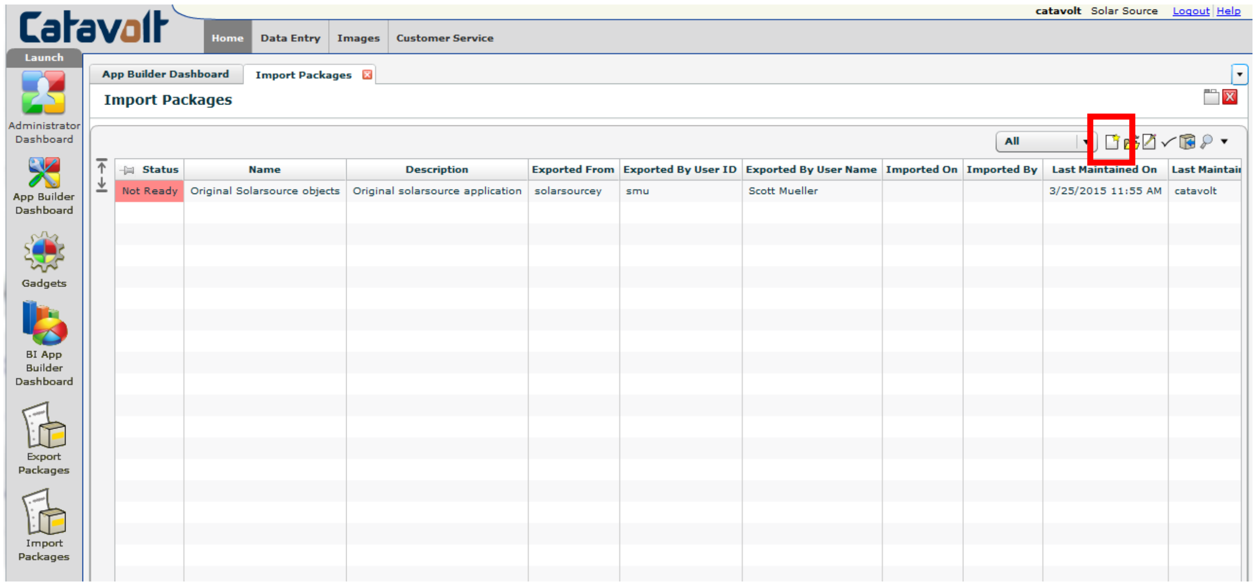
User Profiles are also not exported. While related objects such as Security Roles, Workbenches, etc., are included in Export, they will not contain any assigned users when imported into the target environment.

Import

When performing an Import, you must first create an Import Package. An Import Package defines the contents of the import. Your import ZIP file will be loaded into the system and analyzed. A list of all the objects will be produced, along with a list of issues and warnings that need your attention. After resolving any open issues, you can then perform the import and load the objects into the environment.

Creating Import Packages

To create an Import Package, choose the New menu option.

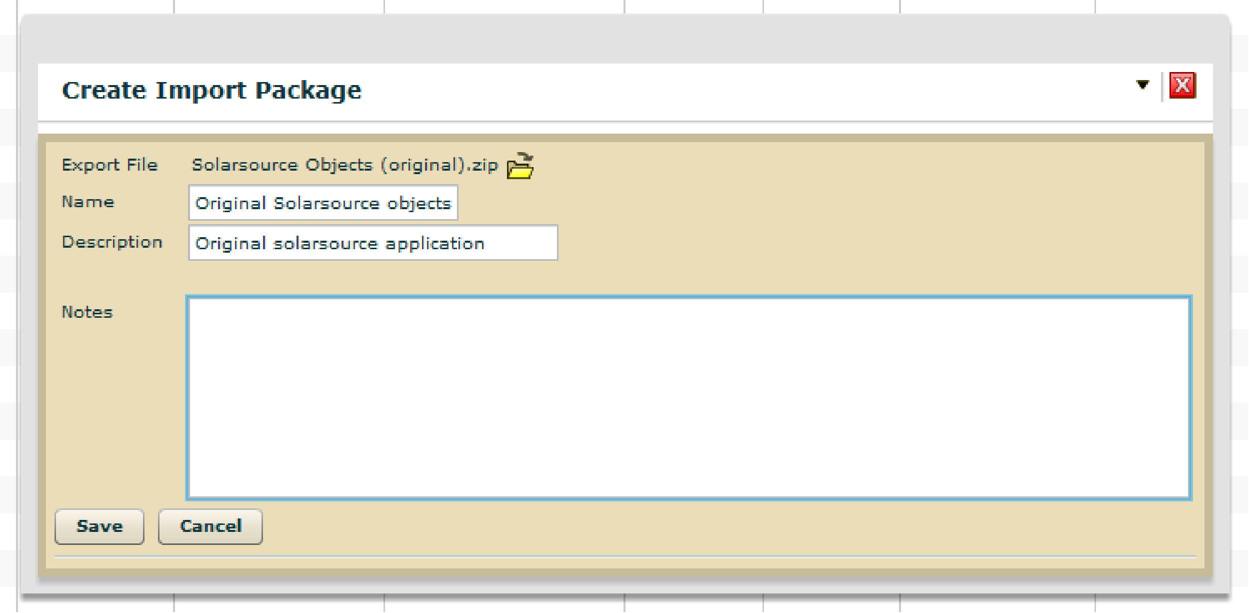


Import Packages

Status	Name	Description	Exported From	Exported By User ID	Exported By User Name	Imported On	Imported By	Last Maintained On	Last Maintai
Not Ready	Original Solarsource objects	Original solarsource application	solarsourcey	smu	Scott Mueller			3/25/2015 11:55 AM	catavolt

Figure 16: The Import Packages list view with the New menu option highlighted

You will be presented with the following window:



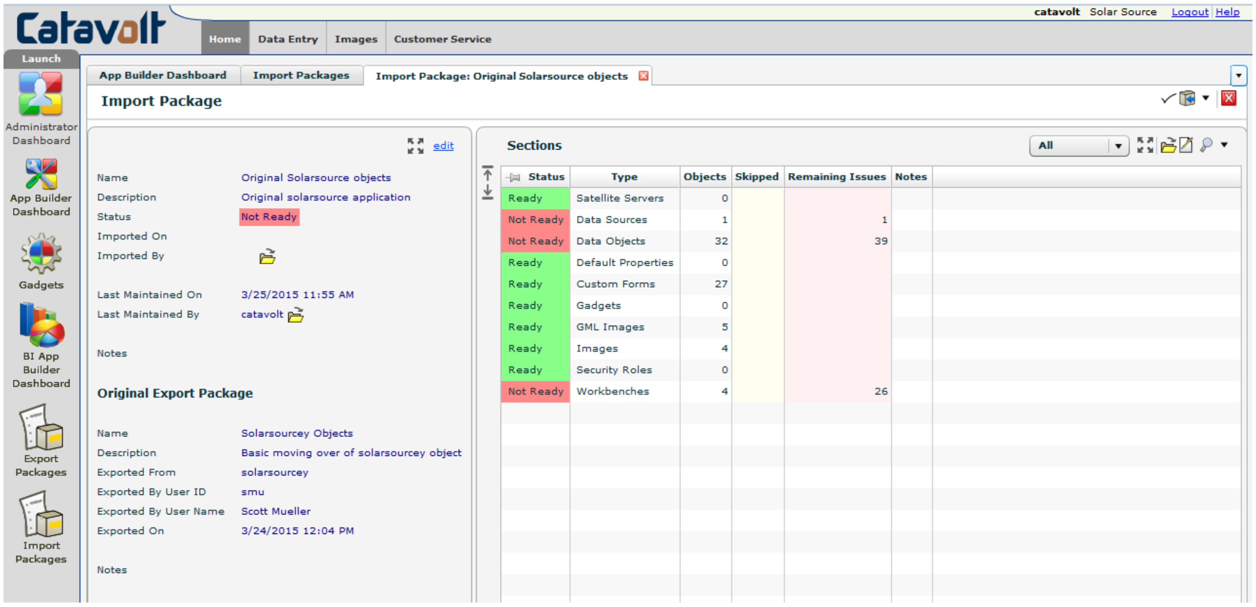
Create Import Package

Export File	Solarsource Objects (original).zip
Name	Original Solarsource objects
Description	Original solarsource application
Notes	

Figure 17: The Create Import Package view

In the Export File, you should select the ZIP file that was produced by the Export process. You may also specify a Name, Description, and Notes for the Import Package. If you leave them blank, they will be defaulted to the values you specified for these fields in the Export Package.

When you press Save, the contents of the zip file will be loaded and analyzed. The Import Package will be created and you will be brought to the following screen which will summarize the objects to be imported and any issues that are currently preventing a successful import.



Import Package

Name	Original Solarsource objects
Description	Original solarsource application
Status	Not Ready
Imported On	
Imported By	
Last Maintained On	3/25/2015 11:55 AM
Last Maintained By	catavolt
Notes	

Original Export Package

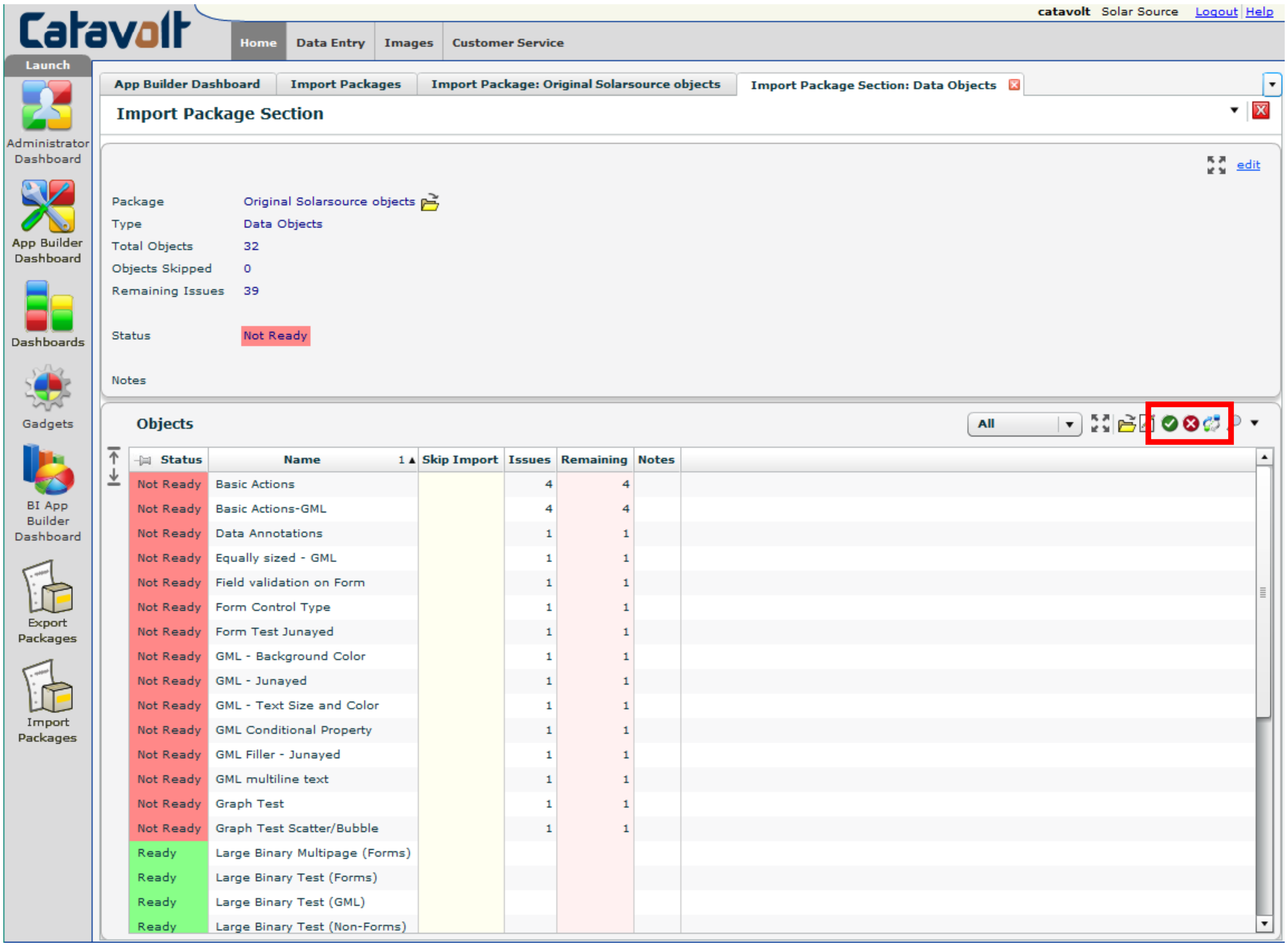
Name	Solarsourcey Objects
Description	Basic moving over of solarsourcey object
Exported From	solarsourcey
Exported By User ID	smu
Exported By User Name	Scott Mueller
Exported On	3/24/2015 12:04 PM
Notes	

Sections

Status	Type	Objects	Skipped	Remaining Issues	Notes
Ready	Satellite Servers	0			
Not Ready	Data Sources	1		1	
Not Ready	Data Objects	32		39	
Ready	Default Properties	0			
Ready	Custom Forms	27			
Ready	Gadgets	0			
Ready	GML Images	5			
Ready	Images	4			
Ready	Security Roles	0			
Not Ready	Workbenches	4		26	

Figure 18: The Import Package detail view

Drilling into a Section, you can see a list of the objects (Data Objects, Data Sources, Workbenches, etc) that will be imported into the system along with the number of total and unresolved issues left. Once all issues for the object have been resolved, the Status of the object will move to Ready. There are also menu actions to Skip Import and Include in Import. Selecting one or more records and pressing the Skip Import button will resolve all issues with a Skip Object resolution and mark the object to not be imported. Selecting one or more records and pressing the Include in Import button will effectively reverse a Skip Import. The object will no longer be marked as skipped and all issues will be reset back to their original state. For Data Source and Data Object types, there is a third action called Reassign Reference. This is a special action that allows you to reassign a reference for multiple records at a time. See **Other Import Menu Options** for more information on this action.



Import Package Section

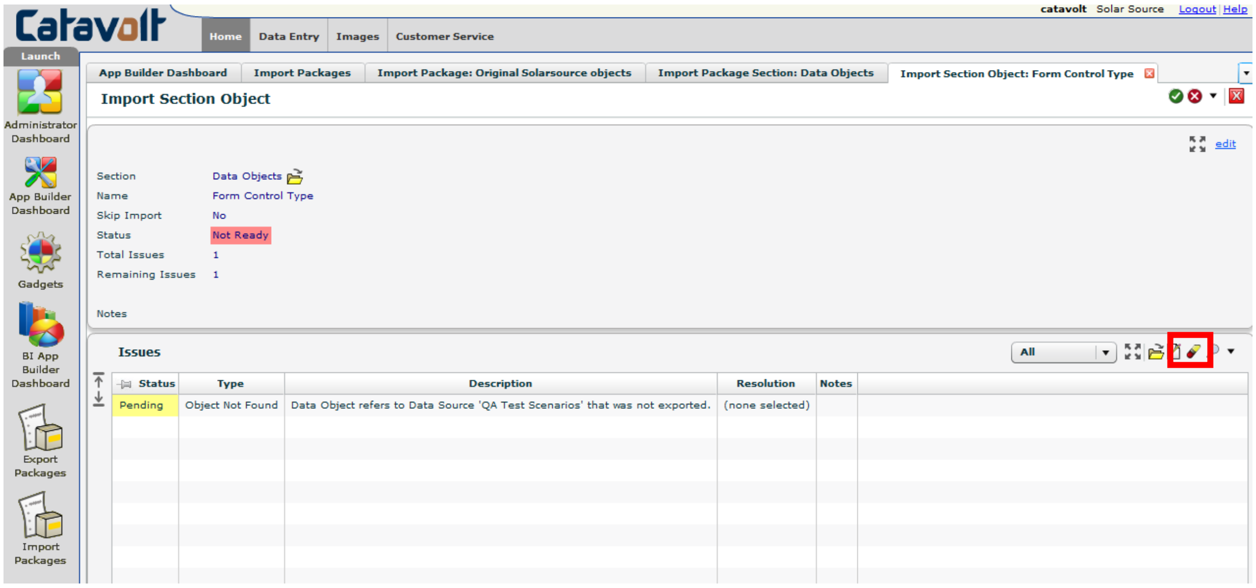
Package	Original Solarsource objects
Type	Data Objects
Total Objects	32
Objects Skipped	0
Remaining Issues	39
Status	Not Ready
Notes	

Objects

Status	Name	Skip Import	Issues	Remaining	Notes
Not Ready	Basic Actions		4	4	
Not Ready	Basic Actions-GML		4	4	
Not Ready	Data Annotations		1	1	
Not Ready	Equally sized - GML		1	1	
Not Ready	Field validation on Form		1	1	
Not Ready	Form Control Type		1	1	
Not Ready	Form Test Junayed		1	1	
Not Ready	GML - Background Color		1	1	
Not Ready	GML - Junayed		1	1	
Not Ready	GML - Text Size and Color		1	1	
Not Ready	GML Conditional Property		1	1	
Not Ready	GML Filler - Junayed		1	1	
Not Ready	GML multiline text		1	1	
Not Ready	Graph Test		1	1	
Not Ready	Graph Test Scatter/Bubble		1	1	
Ready	Large Binary Multipage (Forms)				
Ready	Large Binary Test (Forms)				
Ready	Large Binary Test (GML)				
Ready	Large Binary Test (Non-Forms)				

Figure 19: The Import Package Section detail view with the Include in Import, Skip Import, and Reassign Reference actions highlighted

Drilling into an Object, you can see a list of the Issues preventing the object from being imported into the system along with the current Resolution, if any. Once an Issue has been resolved, the Status of the Issue will move from Pending to Resolved. Sometimes, if an issue has only one possible resolution (for example, a Workbench launcher that refers to a BI Dashboard), the issue may be automatically resolved at the time the ZIP file is loaded and analyzed. There is also a Reset Resolution menu action. Selecting one or more records and pressing the Reset Resolution button will undo a resolution and set its status back to Pending.



Import Section Object

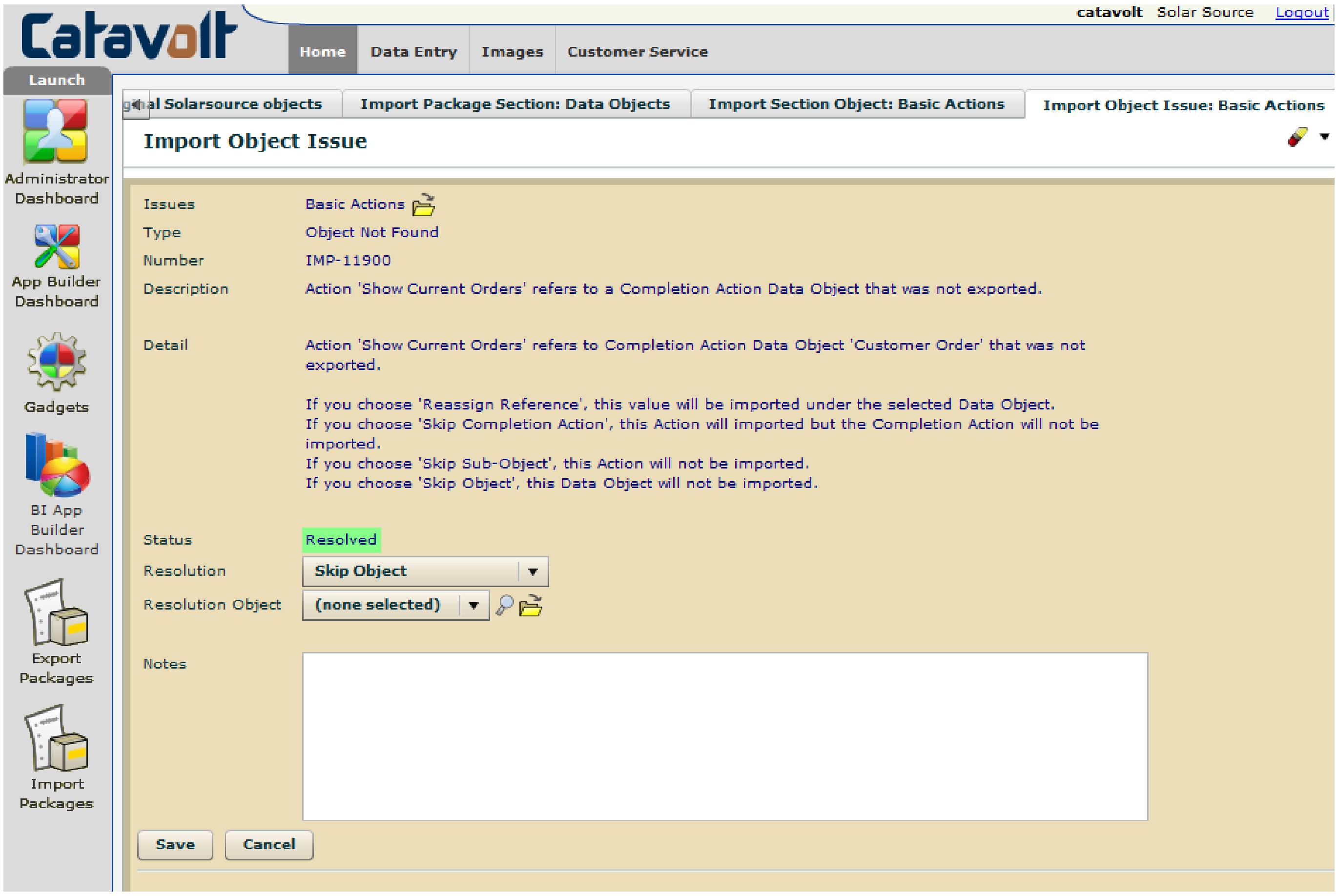
Section	Data Objects
Name	Form Control Type
Skip Import	No
Status	Not Ready
Total Issues	1
Remaining Issues	1
Notes	

Issues

Status	Type	Description	Resolution	Notes
Pending	Object Not Found	Data Object refers to Data Source 'QA Test Scenarios' that was not exported.	(none selected)	

Figure 20: The Import Section Object detail view with the Reset Resolution button highlighted

Drilling into the Issue will show more information about the exact problem. Detail will more fully describe the problem as well as describing each of the available Resolutions and what action will be taken. Some Resolutions such as Skip Object are self contained, while others such as Reassign Reference require more information. For those that require more information, you will need to enter or select a value in the Resolution Object field.



Import Object Issue

Issues	Basic Actions
Type	Object Not Found
Number	IMP-11900
Description	Action 'Show Current Orders' refers to a Completion Action Data Object that was not exported.
Detail	Action 'Show Current Orders' refers to Completion Action Data Object 'Customer Order' that was not exported. If you choose 'Reassign Reference', this value will be imported under the selected Data Object. If you choose 'Skip Completion Action', this Action will imported but the Completion Action will not be imported. If you choose 'Skip Sub-Object', this Action will not be imported. If you choose 'Skip Object', this Data Object will not be imported.
Status	Resolved
Resolution	Skip Object
Resolution Object	(none selected)
Notes	

Figure 21: The Import Object Issue detail view

Below is the current list of issues and possible resolutions that you may encounter when importing. Note that a number of these issues refer to "previously imported objects". When a Data Object, Workbench, etc. is imported, Xalt retains the identity and environment of where the object was originally exported from. This information is used in a couple of important ways. First, when importing an object, Import/Export can recognize that the object has previously been imported into the environment and can give you an option to replace the older version of the object with the new version. Second, if you import an object that has a reference to another object that was not also imported, Import/Export can see that the other object may have already been previously imported and can give you the option to "link up" with the existing object. Finally, when "linking up" with existing objects, Import/Export can determine the identity of sub-objects. For example, if you have a Detail Query Section and you link up with an existing Data Object, Import/Export will also attempt to link up the Rich and Mobile Queries for that Data Object. In cases where there is no existing Query to link up with, Import/Export will use the default Rich/Mobile Query for the Data Object instead.

Connector gateway

IMP-40000 A Connector gateway with the same System URL already exists in this environment.

This issue occurs when you are importing a Connector gateway whose System URL matches the System URL of an existing Connector gateway in the importing environment. You have the following options:

Resolution	Resolution Object	Description
Import and Add	(none)	Import the Connector gateway as a new instance
Skip Object	(none)	Do not Import the Connector gateway. Any Data Sources imported under this Connector gateway will need to be reassigned.

Data Source

IMP-00100 Data Source refers to Connector gateway '${1}' that was not exported.

This issue occurs when you are importing a Data Source and the Connector gateway to which it points is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Existing Connector gateway	Import the Data Source and make it point to the selected Connector gateway from the importing environment
Skip Object	(none)	Do not Import the Data Source. Any Data Objects imported under this Data Source will need to be reassigned.

IMP-00200 Data Source '${1}' is already set to Authenticate External Users.

This issue occurs when you are importing a Data Source that has Authenticate External Users set to Yes, and an existing Data Source in the importing environment already has a Data Source with this value set to Yes. Only one Data Source in an environment can have Authenticate External Users set to Yes. You have the following options:

Resolution	Resolution Object	Description
Skip Setting	(none)	Import the Data Source and set Authenticate External Users to No. The existing Data Source will continue to authenticate external users.

Import Setting	(none)	Import the Data Source and set Authenticate External Users to Yes. The existing Data Source will have its value set to No. The new Data Source will now authenticate external users.
Skip Object	(none)	Do not Import the Data Source. Any Data Objects imported under this Data Source will need to be reassigned.

IMP-00300 Data Source refers to Application '${1}' that is not installed.

This issue occurs when you are importing a Data Source for a particular Application (e.g. Supplier Portal) to which your importing environment is not licensed. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Source. Since your environment is not licensed for the application, Skip Object is automatically set as the Resolution.

IMP-00400 Data Source '${1}' with the same alias already exists in this environment.

This issue occurs when you are importing a Data Source and an existing Data Source already has the same GML/Form Alias. Two Data Sources cannot have the same GML/Form Alias. You have the following options:

Resolution	Resolution Object	Description
Reassign Alias	New Alias value	Import the Data Source and set its GML/Form Alias to the new value entered.
Skip Object	(none)	Do not Import the Data Source. Any Data Objects imported under this Data Source will need to be reassigned.

IMP-00500 Data Source '${1}' with the same Description already exists in this environment.

This issue occurs when you are importing a Data Source and an existing Data Source already has the same Description. This is not an error, but can lead to confusion after the import is complete. You have the following options:

Resolution	Resolution Object	Description
Import Value	(none)	Import the Data Source with its existing Description. Import Value is automatically set as the Resolution but can be changed.
Reassign Reference	New Description value	Import the Data Source and set its Description to the new value entered.
Skip Object	(none)	Do not Import the Data Source. Any Data Objects imported under this Data Source will need to be reassigned.

Data Object

IMP-10000 Data Object refers to Data Source '${1}' that was not exported.

This issue occurs when you are importing a Data Object and the Data Source to which it points is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Existing Data Source	Import the Data Object and make it point to the selected Data Source from the importing environment. Import will verify that the Data Source type (Direct SQL, Salesforce, etc.) matches and the domain class exists in the chosen Data Source.
Skip Object	(none)	Do not Import the Data Object.

IMP-10050 Data Object refers to Domain Class '${1}' which does not exist in this environment and will be skipped.

This issue occurs when you are importing a Data Object and the Domain Class to which it points does not exist in the Data Source you are importing to. This issue can be resolved by Reassigning the Domain Class. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object.

IMP-10100 This Data Object has been previously imported into this environment.

This issue occurs when you are importing a Data Object, have reassigned it to an existing Data Source (see IMP-10000 above), and the Data Source you have chosen already contains a previously imported copy of this Data Object. You have the following options:

Resolution	Resolution Object	Description
Import and Add	(none)	Import the Data Object as a new Data Object. The existing Data Object that was previously imported will be left unchanged.
Import and Overwrite	(none)	Import the Data Object, replacing the existing Data Object that was previously imported.
Skip Object	(none)	Do not Import the Data Object.

IMP-10200 Data Object '${1}' with the same alias already exists in this environment.

This issue occurs when you are importing a Data Object, have reassigned it to an existing Data Source (see IMP-10000 above), and the Data Source you have chosen already contains a Data Object with the same GML/Form Alias. Two Data Objects in the same Data Source cannot have the same GML/Form Alias. You have the following options:

Resolution	Resolution Object	Description
Reassign Alias	New Alias value	Import the Data Object and set its GML/Form Alias to the new value entered.
Skip Object	(none)	Do not Import the Data Object.

IMP-10300 External Resource Prefix '${1}' is already being used by existing Data Objects in this environment.

This issue occurs when you are importing a Data Object and one or more existing Data Objects have the same External Resource Prefix. You have the following options:

Resolution	Resolution Object	Description
Reassign Alias	New External Resource Prefix value	Import the Data Object and set its External Resource Prefix to the new value entered.
Import Value	(none)	Import the Data Object as is. The new Data Object will share the same External Resource Prefix as the existing Data Objects.
Skip Object	(none)	Do not Import the Data Object.

IMP-10400 Exported Key does not match Key of existing Data Objects.

This issue occurs when you are importing a Data Object, an existing Data Object over the same Domain Class already exists in the importing environment, and the key fields do not match. All Data Objects over the same Domain Class must share a common key. You have the following options:

Resolution	Resolution Object	Description
Import Value	(none)	Import the Data Object as is. Existing Data Objects over the same Domain Class will have their key fields updated to match the key of the imported object.
Skip Object	(none)	Do not Import the Data Object.

IMP-10500 Data Object refers to Connection Profile '${1}' that was not exported.

This issue occurs when you are importing a Data Object, have reassigned it to an existing Data Source (see IMP-10000 above), and the Data Source you have chosen does not contain the Connection Profile referenced by this Data Object. You have the following options:

Resolution	Resolution Object	Description
Skip Connection Profile	(none)	Import the Data Object, resetting the Connection Profile property back to its default setting of (data source).
Skip Object	(none)	Do not Import the Data Object.

IMP-10600 Overwritten Data Object '${1}' deleted Property '${2}' is referenced by other Data Objects.

This issue occurs when you are importing a Data Object, have chosen to overwrite an existing Data Object (see IMP-10100 above), and the existing Data Object has a Property that will be deleted as part of the overwrite (because it is not being re-imported with the new Data Object). If there are other existing Data Objects referring to this property, this is considered a dangling reference and will need to be fixed. This issue can only be resolved outside the Import Package. You must go to listed Data Objects, remove or change the reference to another Property, and then choose the 'Recheck Import Package' menu option on the Import Package. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object.

IMP-10700 Property '${1}' does not exist in this environment and will be skipped.

This issue occurs when you are importing a Data Object, have reassigned it to an existing Data Source (see IMP-10000 above), and the Data Object contains a Property that does not exist in the existing Data Source (e.g. the database table is missing a column with the same name as the Property). This issue can only be resolved outside the Import Package. You must go to your back-end system and ensure the missing property is added, then choose the 'Recheck Import Package' menu option on the Import Package. This issue can also be resolved by Reassigning the Domain Class. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object.

IMP-10800 Property '${1}' has a Logical Calculation that does not match the Logical Calculation of existing Data Objects.

This issue occurs when you are importing a Data Object, an existing Data Object over the same Domain Class already exists in the importing environment, and the Logical Calculations for a property do not match. All Data Objects over the same Domain Class must share a common Logical Calculation for properties of the same name. You have the following options:

Resolution	Resolution Object	Description
Import Value	(none)	Import the Data Object as is. Existing Data Objects over the same Domain Class will have their Logical Calculations updated to match the Logical Calculation of the imported object.
Skip Object	(none)	Do not Import the Data Object.

IMP-10900 Property '${1}' already exists in this environment with a different Property Type and will be skipped.

This issue occurs when you are importing a Data Object, an existing Data Object over the same Domain Class already exists in the importing environment, and the Types for a property do not match. All Data Objects over the same Domain Class must share a common Property Type (Physical or Logical) for properties of the same name. This issue can only be resolved outside the Import Package. You must remove or ensure the properties are consistent, then choose the 'Recheck Import Package' menu option on the Import Package. This issue can also be resolved by Reassigning the Domain Class. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object. Since different Property Types are not allowed, Skip Object is automatically set as the Resolution.

IMP-11000 Additional Query Scope '${1}' refers to object '${2}' which does not exist in this environment and will be skipped.

This issue occurs when you are importing a Data Object, have reassigned it to an existing Data Source (see IMP-10000 above), and the Domain Class defined in an Additional Query Scope does not exist in the Data Source. This issue can only be resolved outside the Import Package. You must go to your back-end system and ensure the missing domain class is added, then choose the 'Recheck Import Package' menu option on the Import Package. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object. Since missing Domain Classes are not allowed, Skip Object is automatically set as the Resolution.

IMP-11100 Additional Query Scope '${1}' already exists in this environment with a different Domain Class.

This issue occurs when you are importing a Data Object, an existing Data Object over the same Domain Class already exists in the importing environment, and an Additional Query Scope with the same Name exists but the domain classes do not match. All Data Objects over the same Domain Class must share a common domain class for Additional Query Scopes of the same name. You have the following options:

Resolution	Resolution Object	Description
Import Value	(none)	Import the Data Object as is. Existing Data Objects over the same Domain Class will have their Additional Query Scopes updated to match the Additional Query Scope of the imported object.
Skip Object	(none)	Do not Import the Data Object.

IMP-11200 Additional Query Scope '${1}' refers to property '${2}' which does not exist in this environment and will be skipped.

This issue occurs when you are importing a Data Object, have reassigned it to an existing Data Source (see IMP-10000 above), and the Data Object contains an Additional Query Scope with a Join Property that does not exist in the existing Data Source (e.g. the database table is missing a column with the same name as the Property). This issue can only be resolved outside the Import Package. You must go to your back-end system and ensure the missing property is added, then choose the 'Recheck Import Package' menu option on the Import Package. This issue can also be resolved by Reassigning the Domain Class. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object. Since missing Properties are not allowed, Skip Object is automatically set as the Resolution.

IMP-11300 Lookup '${1}' refers to a Data Object that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains an Object Lookup that refers to a Data Object that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Data Object	Import the Data Object and set its Object Lookup to point to the selected Data Object.
Skip Sub-Object	(none)	Import the Data Object but skip the Object Lookup.
Skip Object	(none)	Do not Import the Data Object.

IMP-11400 Overwritten Data Object '${1}' deleted Launcher '${2}' is referenced by other Data Objects.

This issue occurs when you are importing a Data Object, have chosen to overwrite an existing Data Object (see IMP-10100 above), and the existing Data Object has a Detail or Query Launcher that will be deleted as part of the overwrite (because it is not being re-imported with the new Data Object). If there are other existing Data Objects referring to this Launcher, this is considered a dangling reference and will need to be fixed. This issue can only be resolved outside the Import Package. You must go to listed Data Objects, remove or change the reference to another Launcher, and then choose the 'Recheck Import Package' menu option on the Import Package. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object.

IMP-11500 Overwritten Data Object '${1}' deleted Detail '${2}' is referenced by other Data Objects.

This issue occurs when you are importing a Data Object, have chosen to overwrite an existing Data Object (see IMP-10100 above), and the existing Data Object has a Detail that will be deleted as part of the overwrite (because it is not being re-imported with the new Data Object). If there are other existing Data Objects referring to this Detail, this is considered a dangling reference and will need to be fixed. This issue can only be resolved outside the Import Package. You must go to listed Data Objects, remove or change the reference to another Detail, and then choose the 'Recheck Import Package' menu option on the Import Package. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object.

IMP-11600 Detail '${1}' has a Query Section '${2}' that refers to a Data Object that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains a Detail Query Section that refers to a Data Object that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Data Object	Import the Data Object and set its Detail Query Section to point to the selected Data Object.
Skip Query Section	(none)	Import the Data Object but skip the Detail Query Section
Skip Object	(none)	Do not Import the Data Object.

IMP-11700 Overwritten Data Object '${1}' deleted Query '${2}' is referenced by other Data Objects.

This issue occurs when you are importing a Data Object, have chosen to overwrite an existing Data Object (see IMP-10100 above), and the existing Data Object has a Query that will be deleted as part of the overwrite (because it is not being re-imported with the new Data Object). If there are other existing Data Objects referring to this Query, this is considered a dangling reference and will need to be fixed. This issue can only be resolved outside the Import Package. You must go to listed Data Objects, remove or change the reference to another Query, and then choose the 'Recheck Import Package' menu option on the Import Package. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object.

IMP-11800 Overwritten Data Object '${1}' deleted Action '${2}' is referenced by other Data Objects.

This issue occurs when you are importing a Data Object, have chosen to overwrite an existing Data Object (see IMP-10100 above), and the existing Data Object has an Action that will be deleted as part of the overwrite (because it is not being re-imported with the new Data Object). If there are other existing Data Objects referring to this Action, this is considered a dangling reference and will need to be fixed. This issue can only be resolved outside the Import Package. You must go to listed Data Objects, remove or change the reference to another Action, and then choose the 'Recheck Import Package' menu option on the Import Package. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Data Object.

IMP-11900 Action '${1}' refers to a Completion Action Data Object that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains an Action with a Completion Action that refers to a Data Object that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Data Object	Import the Data Object and set its Completion Action to point to the selected Data Object.
Skip Completion Action	(none)	Import the Data Object but skip the Completion Action for the Action.
Skip Sub-Object	(none)	Import the Data Object but skip the Action.
Skip Object	(none)	Do not Import the Data Object.

IMP-12000 Action '${1}' refers to a Completion Action Data Object Action that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains an Action with a Completion Action that refers to an Action that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Completion Action	(none)	Import the Data Object but skip the Completion Action for the Action.
Skip Sub-Object	(none)	Import the Data Object but skip the Action.
Skip Object	(none)	Do not Import the Data Object.

IMP-12100 Action '${1}' refers to a Workflow Completion Action Data Object that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains an Action with a Workflow Completion Action that refers to a Data Object that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Data Object	Import the Data Object and set its Workflow Completion Action to point to the selected Data Object.
Skip Workflow Completion Action	(none)	Import the Data Object but skip the Workflow Completion Action for the Action.
Skip Sub-Object	(none)	Import the Data Object but skip the Action.
Skip Object	(none)	Do not Import the Data Object.

IMP-12200 Action '${1}' refers to a Workflow Completion Action Data Object Action that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains an Action with a Workflow Completion Action that refers to an Action that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Workflow Completion Action	(none)	Import the Data Object but skip the Workflow Completion Action for the Action.
Skip Sub-Object	(none)	Import the Data Object but skip the Action.
Skip Object	(none)	Do not Import the Data Object.

IMP-12300 Action '${1}' refers to a Completion Action Dashboard that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains an Action with a Completion Action that refers to a Dashboard that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Dashboard	Import the Data Object and set its Completion Action to point to the selected Dashboard.
Skip Completion Action	(none)	Import the Data Object but skip the Completion Action for the Action.
Skip Sub-Object	(none)	Import the Data Object but skip the Action.
Skip Object	(none)	Do not Import the Data Object.

IMP-12400 Action '${1}' refers to a Workflow Completion Action Dashboard that was not exported.

This issue occurs when you are importing a Data Object and the Data Object contains an Action with a Workflow Completion Action that refers to a Dashboard that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Dashboard	Import the Data Object and set its Workflow Completion Action to point to the selected Dashboard.
Skip Workflow Completion Action	(none)	Import the Data Object but skip the Workflow Completion Action for the Action.
Skip Sub-Object	(none)	Import the Data Object but skip the Action.
Skip Object	(none)	Do not Import the Data Object.

IMP-12500 Logical Property '${1}' does not exist in this environment for existing Data Objects with the same Domain Class.

This issue occurs when you are importing a Data Object, an existing Data Object over the same Domain Class already exists in the importing environment, and a new Logical Property is being imported that does not exist on the existing Data Object. All Data Objects over the same Domain Class must share a common set of Logical Properties. You have the following options:

Resolution	Resolution Object	Description

Import Value	(none)	Import the Data Object as is. Existing Data Objects over the same Domain Class will add the new Logical Property of the imported object.
Skip Object	(none)	Do not Import the Data Object.

IMP-12600 Additional Query Scope '${1}' does not exist in this environment for existing Data Objects with the same Domain Class.

This issue occurs when you are importing a Data Object, an existing Data Object over the same Domain Class already exists in the importing environment, and a new Additional Query Scope is being imported that does not exist on the existing Data Object. All Data Objects over the same Domain Class must share a common set of Additional Query Scopes. You have the following options:

Resolution	Resolution Object	Description
Import Value	(none)	Import the Data Object as is. Existing Data Objects over the same Domain Class will add the new Additional Query Scope of the imported object.
Skip Object	(none)	Do not Import the Data Object.

Workbench

IMP-20000 This Workbench has been previously imported into this environment.

This issue occurs when you are importing a Workbench into an environment that already contains a previously imported copy of this Workbench. You have the following options:

Resolution	Resolution Object	Description
Import and Add	(none)	Import the Workbench as a new Workbench. The existing Workbench that was previously imported will be left unchanged.
Import and Overwrite	(none)	Import the Workbench, replacing the existing Workbench that was previously imported.
Skip Object	(none)	Do not Import the Workbench.

IMP-20100 Workbench '${1}' with the same alias already exists in this environment.

This issue occurs when you are importing a Workbench and another Workbench already contains the same GML/Form Alias. Two Workbenches in the same environment cannot have the same GML/Form Alias. You have the following options:

Resolution	Resolution Object	Description
Reassign Alias	New Alias value	Import the Workbench and set its GML/Form Alias to the new value entered.
Skip Object	(none)	Do not Import the Workbench.

IMP-20200 Workbench Launcher '${1}' refers to a BI Object and will be skipped.

This issue occurs when you are importing a Workbench that contains a Dashboard Launcher that refers to a BI Object (BI Objects are not supported by Import/Export). You have the following options:

Resolution	Resolution Object	Description
Skip Sub-Object	(none)	Import the Workbench but skip the Workbench Launcher. Since BI Objects are not supported by Import/Export, Skip Sub-Object is automatically set as the Resolution.
Skip Object	(none)	Do not Import the Workbench.

IMP-20300 Workbench Launcher '${1}' refers to a Data Object that was not exported.

This issue occurs when you are importing a Workbench and a Launcher refers to a Data Object that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Data Object	Import the Workbench and set its Launcher to point to the selected Data Object.
Skip Sub-Object	(none)	Import the Workbench but skip the Launcher.
Skip Object	(none)	Do not Import the Workbench.

IMP-20400 Workbench Launcher '${1}' refers to an Action that was not exported and will be skipped.

This issue occurs when you are importing a Workbench and a Launcher refers to an Action that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Sub-Object	(none)	Import the Workbench but skip the Launcher. Skip Sub-Object is automatically set as the Resolution.
Skip Object	(none)	Do not Import the Workbench.

IMP-20500 Workbench Launcher '${1}' refers to a Standard Action that does not exist in this environment and will be skipped.

This issue occurs when you are importing a Workbench and a Launcher refers to a standard action (e.g. Change Password, App Builder Dashboard) that does not exist in the importing environment. You have the following options:

Resolution	Resolution Object	Description
Skip Sub-Object	(none)	Import the Workbench but skip the Launcher. Skip Sub-Object is automatically set as the Resolution.
Skip Object	(none)	Do not Import the Workbench.

IMP-20600 Workbench Launcher '${1}' refers to a Workbench that was not exported and will be skipped.

This issue occurs when you are importing a Workbench and a Launcher refers to a Workbench that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Sub-Object	(none)	Import the Workbench but skip the Launcher. Skip Sub-Object is automatically set as the Resolution.
Skip Object	(none)	Do not Import the Workbench.

IMP-20700 Workbench Launcher '${1}' refers to a Dashboard that was not exported.

This issue occurs when you are importing a Workbench and a Launcher refers to a Dashboard that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Dashboard	Import the Workbench and set its Launcher to point to the selected Dashboard.
Skip Sub-Object	(none)	Import the Workbench but skip the Launcher.
Skip Object	(none)	Do not Import the Workbench.

Gadget

IMP-25000 Gadget '${1}' contains a launcher '${2}' that refers to an Action that was not exported and will be skipped.

This issue occurs when you are importing a Gadget that refers to an Action that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Gadget. Skip Object is automatically set as the Resolution.

IMP-25100 Gadget '${1}' contains a launcher '${2}' that refers to a BI Object and will be skipped.

This issue occurs when you are importing a Gadget that refers to a BI Object (BI Objects are not supported by Import/Export). You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Gadget. Skip Object is automatically set as the Resolution.

IMP-25200 Gadget '${1}' contains a launcher '${2}' that refers to a Data Object that was not exported.

This issue occurs when you are importing a Gadget that refers to a Data Object that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Data Object	Import the Gadget and set its Launcher to point to the selected Data Object.
Skip Object	(none)	Do not Import the Workbench.

IMP-25300 Gadget '${1}' contains a launcher '${2}' that refers to a Standard Action that does not exist in this environment and will be skipped.

This issue occurs when you are importing a Gadget that refers to a standard action (e.g. Change Password, App Builder Dashboard) that does not exist in the importing environment. You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Gadget. Skip Object is automatically set as the Resolution.

IMP-25400 Gadget '${1}' contains a launcher '${2}' that refers to a Workbench that was not exported and will be skipped.

This issue occurs when you are importing a Gadget that refers to a Workbench that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Gadget. Skip Object is automatically set as the Resolution.

IMP-25500 Gadget '${1}' contains a launcher '${2}' that refers to a Dashboard that was not exported.

This issue occurs when you are importing a Gadget that refers to a Dashboard that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Object	(none)	Do not Import the Gadget. Skip Object is automatically set as the Resolution.

Security Role

IMP-30000 A Security Role with the same name already exists in this environment.

This issue occurs when you are importing a Security Role whose name matches the name of an existing Security Role in the importing environment. You have the following options:

Resolution	Resolution Object	Description
Import and Overwrite	(none)	Import the Security Role, replacing the existing Security Role.
Skip and Reassign	(none)	Do not Import the Security Role and replace all references to point to the existing Security Role.

IMP-30100 Conflicting Role ${1} was not exported and will be skipped.

This issue occurs when you are importing a Security Role and a Conflicting Role refers to a Security Role that is not being imported (it is not in the Import Package). You have the following options:

Resolution	Resolution Object	Description
Skip Sub-Object	(none)	Import the Security Role but skip the Conflicting Role. Skip Sub-Object is automatically set as the Resolution.

IMP-30200 Role Workbench '${1}' was not exported and will be skipped.

This issue occurs when you are importing a Security Role and a Role Workbench refers to a Workbench that is not being imported (it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Skip Sub-Object	(none)	Import the Security Role but skip the Conflicting Role. Skip Sub-Object is automatically set as the Resolution.

IMP-30300 One or more Users are assigned to Conflicting Security Roles '${1}' and '${2}'.

This issue occurs when you are importing one or more Security Roles with Conflicting Roles. If these Security Roles were to be imported, one or more User Profiles would be a member of both Conflicting Roles. This issue can only be resolved outside the Import Package. You must go to listed users and remove them from one of the two Conflicting roles, then choose the 'Recheck Import Package' menu option on the Import Package. You have the following options:

Resolution	Resolution Object	Description

Dashboard

IMP-35000 This Dashboard has been previously imported into this environment.

This issue occurs when you are importing a Dashboard into an environment that already contains a previously imported copy of this Dashboard. You have the following options:

Resolution	Resolution Object	Description
Import and Add	(none)	Import the Dashboard as a new Dashboard. The existing Dashboard that was previously imported will be left unchanged.
Import and Overwrite	(none)	Import the Dashboard, replacing the existing Dashboard that was previously imported.
Skip Object	(none)	Do not Import the Dashboard.

IMP-35100 Dashboard '${1}' with the same alias already exists in this environment.

This issue occurs when you are importing a Dashboard and another Dashboard already contains the same GML/Form Alias. Two Dashboards in the same environment cannot have the same GML/Form Alias. You have the following options:

Resolution	Resolution Object	Description
Reassign Alias	New Alias value	Import the Dashboard and set its GML/Form Alias to the new value entered.
Skip Object	(none)	Do not Import the Dashboard.

IMP-35200 **Dashboard '${1}' has a Query Section '${2}' that refers to a Data Object that was not exported.**

This issue occurs when you are importing a Dashboard and the Dashboard contains a Dashboard Query Section that refers to a Data Object that is not being imported (either it is not in the Import Package or it is being skipped). You have the following options:

Resolution	Resolution Object	Description
Reassign Reference	Previously imported Data Object	Import the Dashboard and set its Dashboard Query Section to point to the selected Data Object.
Skip Query Section	(none)	Import the Dashboard but skip the Detail Query Section
Skip Object	(none)	Do not Import the Dashboard.

Default Property

IMP-50000 **A Default Property with the same name already exists in this environment.**

This issue occurs when you are importing a Default Property whose name matches the name of an existing Default Property in the importing environment. You have the following options:

Resolution	Resolution Object	Description
Import and Overwrite	(none)	Import the Default Property, replacing the existing Default Property. If both Default Properties have the same value, Import and Overwrite will automatically be set as the Resolution.
Skip Object	(none)	Do not Import the Default Property.

Custom Forms

IMP-51000 **A custom form with the same name already exists in this environment.**

This issue occurs when you are importing a Custom Form or Form Image whose name matches the name of an existing Custom Form or Form Image in the importing environment. You have the following options:

Resolution	Resolution Object	Description
Import and Overwrite	(none)	Import the Form/Image, replacing the existing Form/Image.
Skip Object	(none)	Do not Import the Form/Image.

Images

IMP-52000 An image with the same name already exists in this environment.

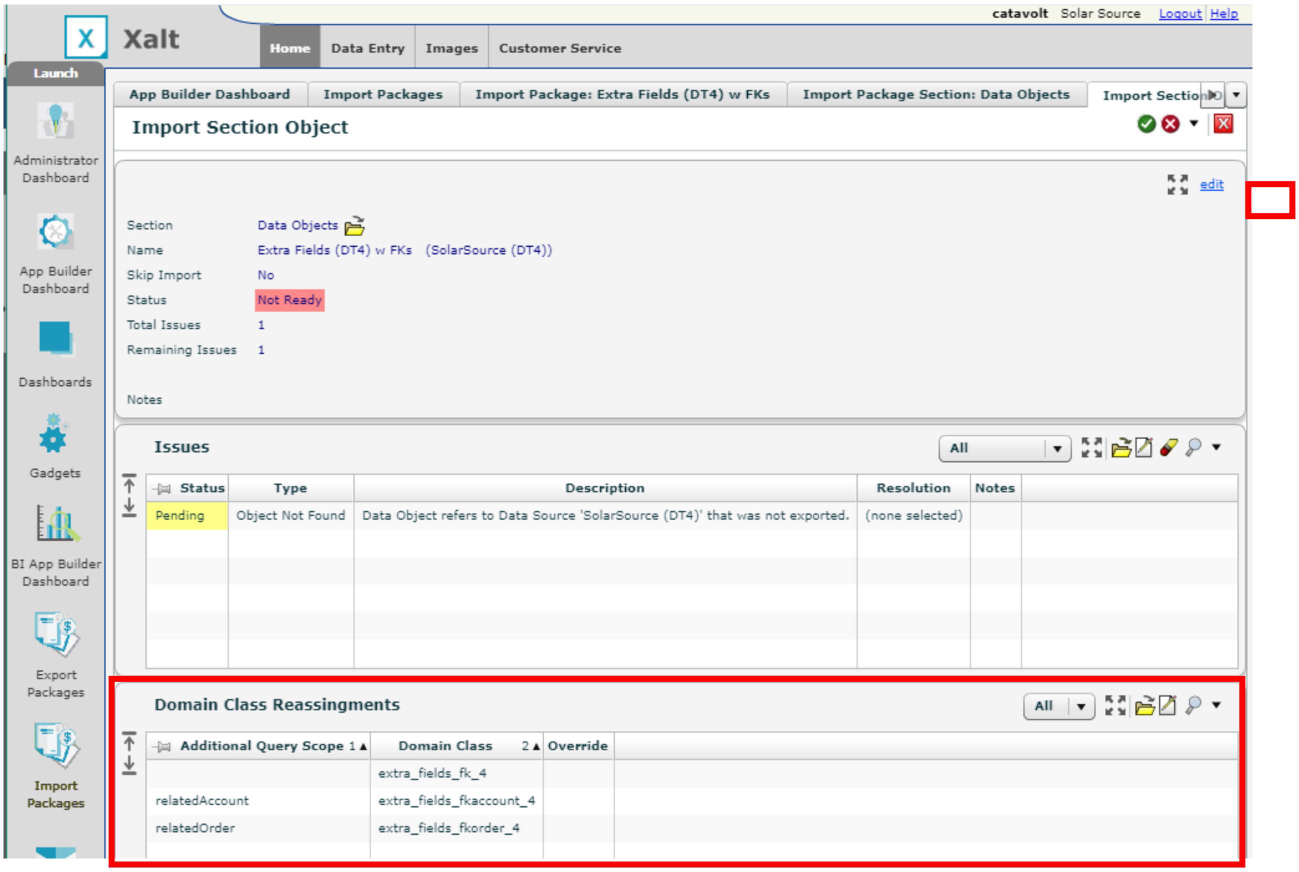
This issue occurs when you are importing an Image whose name matches the name of an existing Image in the importing environment. You have the following options:

Resolution	Resolution Object	Description
Import and Overwrite	(none)	Import the Image, replacing the existing Image.
Skip Object	(none)	Do not Import the Image.

Reassigning Domain Classes

When importing Data Objects, you may need to change to a different a domain class during the Import process. This is typically needed when a Production database has a different naming convention for tables than a Development database, or if are Importing a standard Xalt application and have naming clashes with the tables in your database.

We have added a section called Domain Class Reassignment on the Import Section Object details for Data Object. It will contain a record for the base Data Object along with a record for each Additional Query Scope. You can reassign the domain class for each value independently. Leaving the Override blank will import the Data Object using the same Domain Class it was exported with.



Import Section Object

Section	Data Objects
Name	Extra Fields (DT4) w FKs (SolarSource (DT4))
Skip Import	No
Status	Not Ready
Total Issues	1
Remaining Issues	1
Notes	

Issues

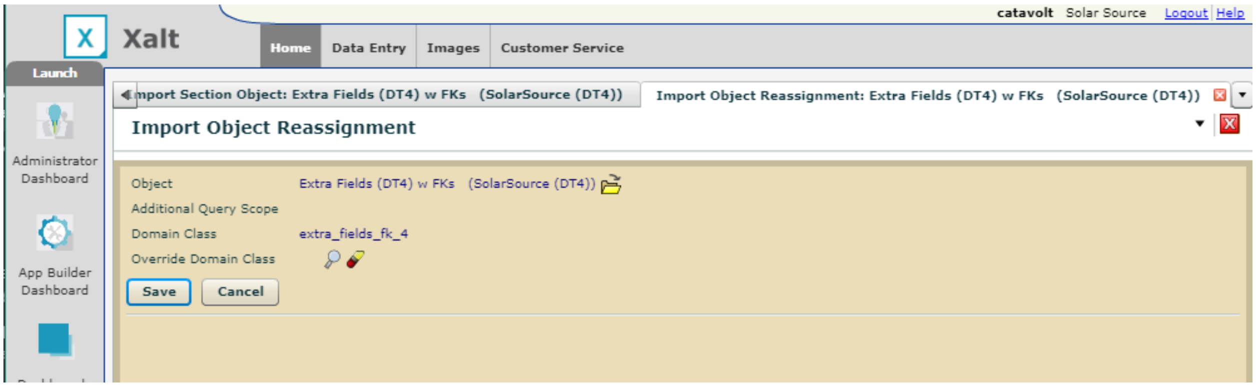
Status	Type	Description	Resolution	Notes
Pending	Object Not Found	Data Object refers to Data Source 'SolarSource (DT4)' that was not exported.	(none selected)	

Domain Class Reassignments

Additional Query Scope 1 ▲	Domain Class 2 ▲	Override
	extra_fields_fk_4	
relatedAccount	extra_fields_fkaccount_4	
relatedOrder	extra_fields_fkorder_4	

Figure 22: The Import Section Object details view for a Data Object with the Domain Class Reassignments section highlighted

Opening a Domain Class Reassignment record will allow you to select a new Override Domain Class, or reset the Override back to blank:


Xalt
catavolt Solar Source Logout | Help
Home
Data Entry
Images
Customer Service
Launch
Administrator Dashboard
App Builder Dashboard
Import Section Object: Extra Fields (DT4) w FKs (SolarSource (DT4))
Import Object Reassignment: Extra Fields (DT4) w FKs (SolarSource (DT4))
Save
Cancel

Import Object Reassignment

Object	Extra Fields (DT4) w FKs (SolarSource (DT4))
Additional Query Scope	
Domain Class	extra_fields_fk_4
Override Domain Class	

Figure 23: The Import Object Reassignment detail view

Note that you can only run reassignment here if you have previously reassigned this Data Object to a new Data Source. If you are importing the entire Data Source, you should simply import the Data Object as-is and then do the normal runtime Domain Class Reassignment, as the runtime reassignment gives you more flexibility.

Reassigning the base domain class for a Data Object will NOT automatically reassign the same domain class for other Data Objects being imported. There is no "common" issue solution for reassignment, each Data Object will have to be reassigned individually.

However, reassigning a domain class for an Additional Query Scope will also change the domain class reassignment on corresponding Additional Query Scopes on all other Data Objects over the same domain class being imported (as all Data Objects over the same Domain Class must share the same Additional Query Scope). This includes any subsequent reassigning of Data Objects to this new domain class. Below is a list of what Xalt will do for specific import scenarios:

Reassign primary Domain Class on a Data Object

Check if any other Data Objects attached to the same Data Source exist over the same Domain Class. If so, take all their Additional Query Scope Domain Class Reassignments and assign them to this Data Object

If no matching Data Objects exist, do not reset Additional Query Scope Domain Class Reassignments for the Data Object.

Reassign Additional Query Scope Domain Class on a Data Object

Check if any other Data Objects attached to the same Data Source exist over the same Domain Class. If so, change their Additional Query Scope Domain Class Reassignments to match the value changed on this Data Object

Reassign Reference for a Data Object to an existing Data Source

Reset all Domain Class Reassignments for the Data Object back to empty

Check if any other Data Objects attached to the same Data Source exist over the same Domain Class. If so, take all their Additional Query Scope Domain Class Reassignments and assign them to this Data Object

Reset Reassign a Data Object to an existing Data Source Issue back to (no selection) or Skip Object

Reset all Domain Class Reassignments for the Data Object back to empty

Run Skip Import action on a Data Object

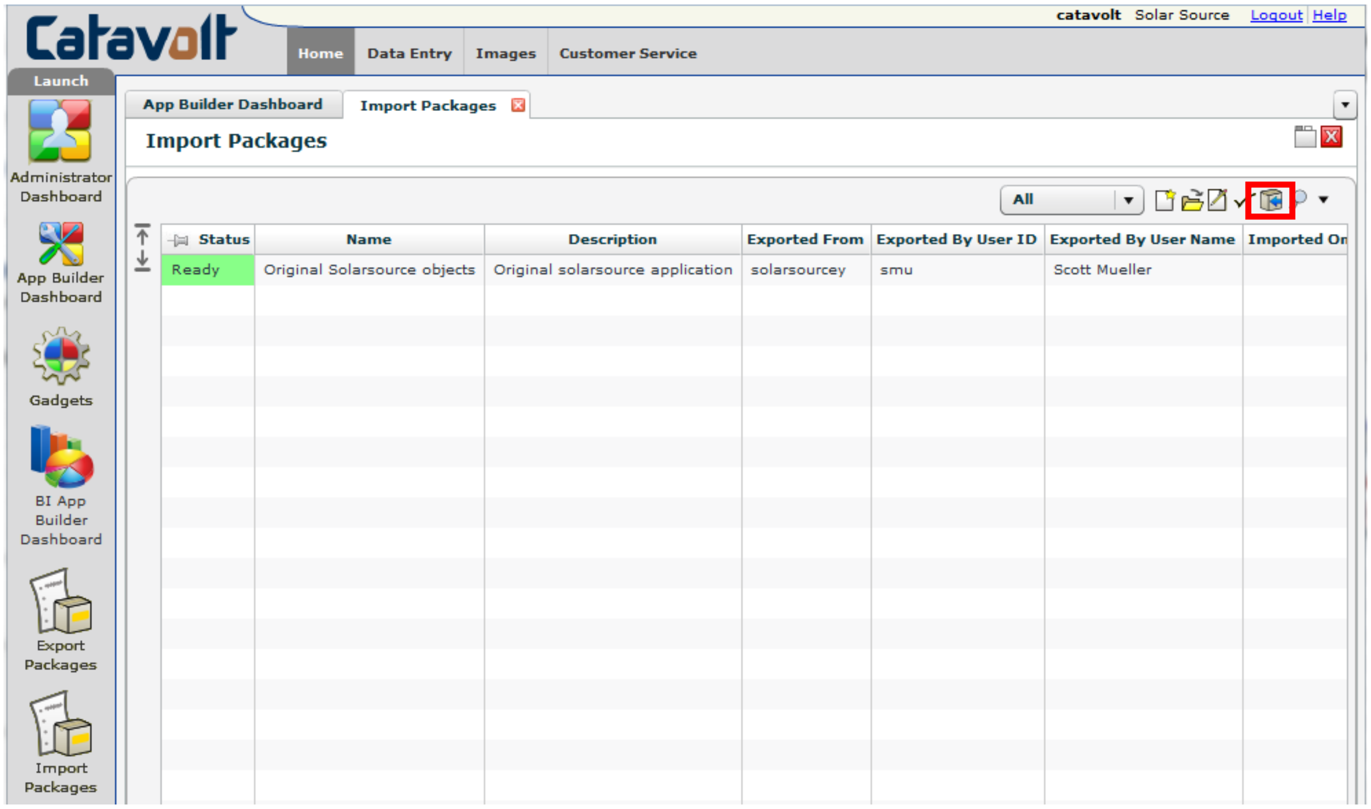
Reset all Domain Class Reassignments for the Data Object back to empty

After reassignment of a Domain Class, Xalt will re-run the standard import checks to get issues as we would for normal import of a Data Object to a different Data Source. The same issues with Object Structure synching apply as for Data Object / Additional Query Scope sections above.

Note that reassignment during import is less flexible than at runtime, as the Import process will force all issues to be resolved before the import takes place to ensure correctness (as opposed to reassignment at runtime where you are allowed to have a "broken" object at the end of the process).

Performing an Import

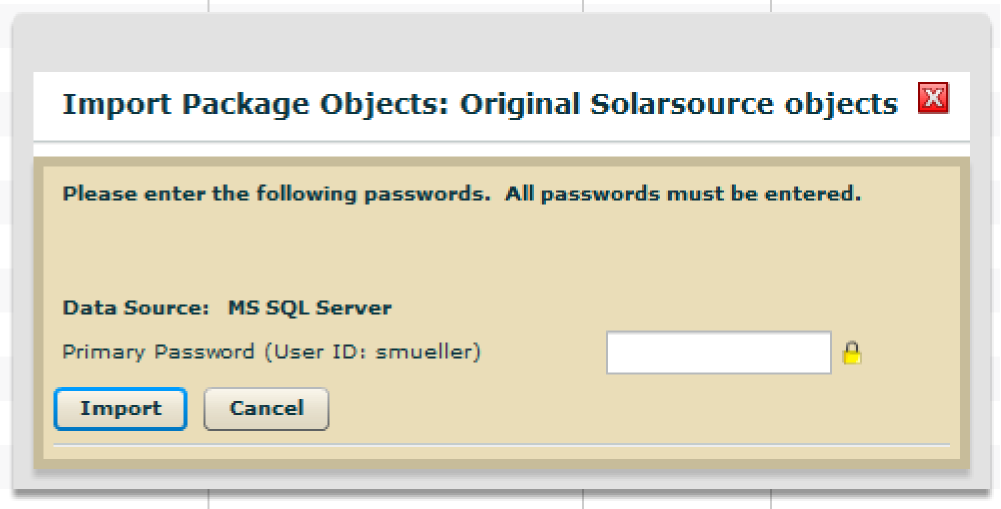
When all issues have been resolved, your Import Package will show a Status of Ready. You can then choose the Import Objects menu action to begin the Import.



Status	Name	Description	Exported From	Exported By User ID	Exported By User Name	Imported Or
Ready	Original Solarsource objects	Original solarsource application	solarsourcey	smu	Scott Mueller	

Figure 24: The Import Package list view with the Import Objects button highlighted

When you choose the menu action, you will be presented with one of the following dialogs. If you are importing objects that have passwords associated with them (e.g. Data Sources, Connector gateways, etc.), you will be required to re-enter the passwords for these objects at the following screen before the Import can commence:



Import Package Objects: Original Solarsource objects

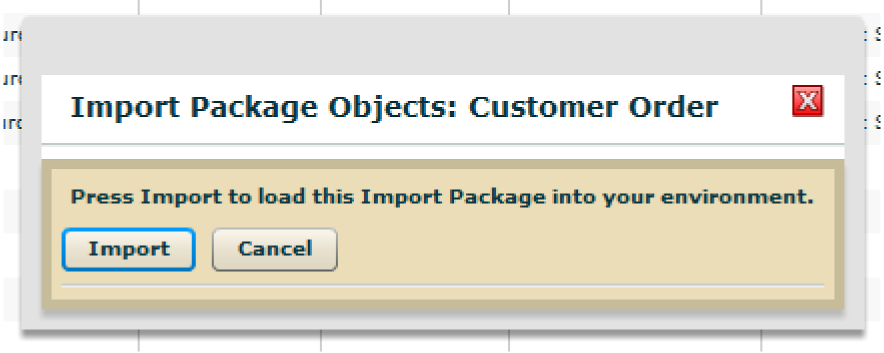
Please enter the following passwords. All passwords must be entered.

Data Source: MS SQL Server

Primary Password (User ID: smueller)

Figure 25: The Import Package Objects prompt view

If you are not importing objects that require a password, you will see the following confirmation prompt

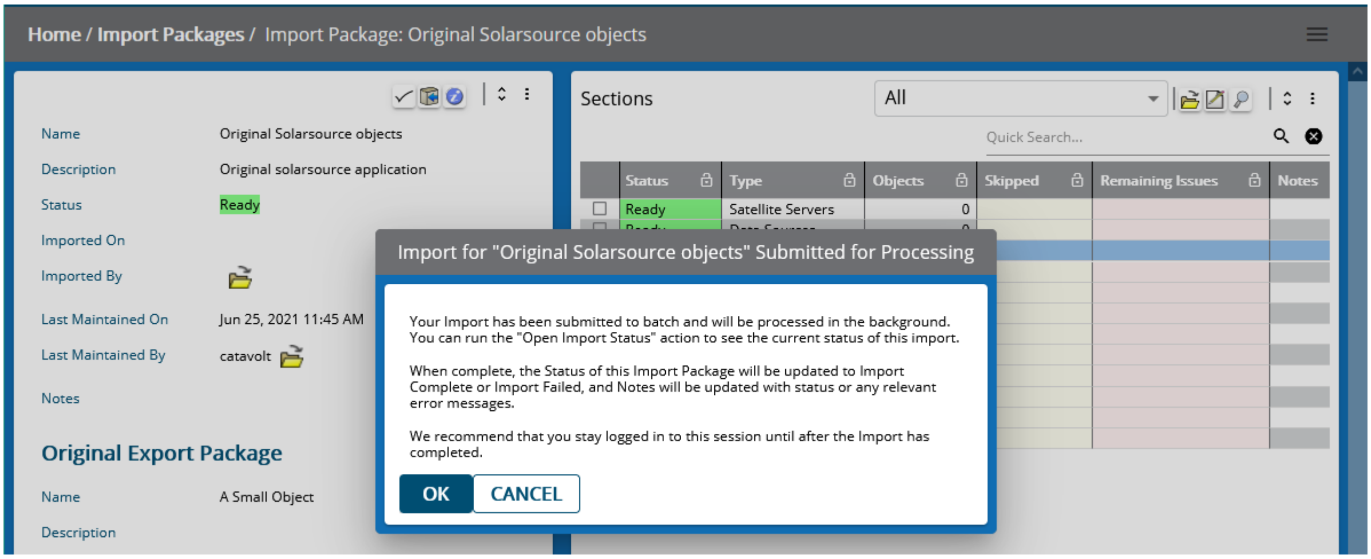


Import Package Objects: Customer Order

Press Import to load this Import Package into your environment.

Figure 26: The Import Package Objects prompt view

As the import process can take significant time, pressing the Import button will submit the import process to run in the background instead of running it inline. When pressing the Import button, you will be presented with the following dialog:



Home / Import Packages / Import Package: Original Solarsource objects

Name: Original Solarsource objects
Description: Original solarsource application
Status: Ready
Imported On:
Imported By:
Last Maintained On: Jun 25, 2021 11:45 AM
Last Maintained By: catavolt
Notes:

Original Export Package

Name: A Small Object
Description:

Sections: All

Status	Type	Objects	Skipped	Remaining Issues	Notes
Ready	Satellite Servers	0			

Import for "Original Solarsource objects" Submitted for Processing

Your Import has been submitted to batch and will be processed in the background. You can run the "Open Import Status" action to see the current status of this import.

When complete, the Status of this Import Package will be updated to Import Complete or Import Failed, and Notes will be updated with status or any relevant error messages.

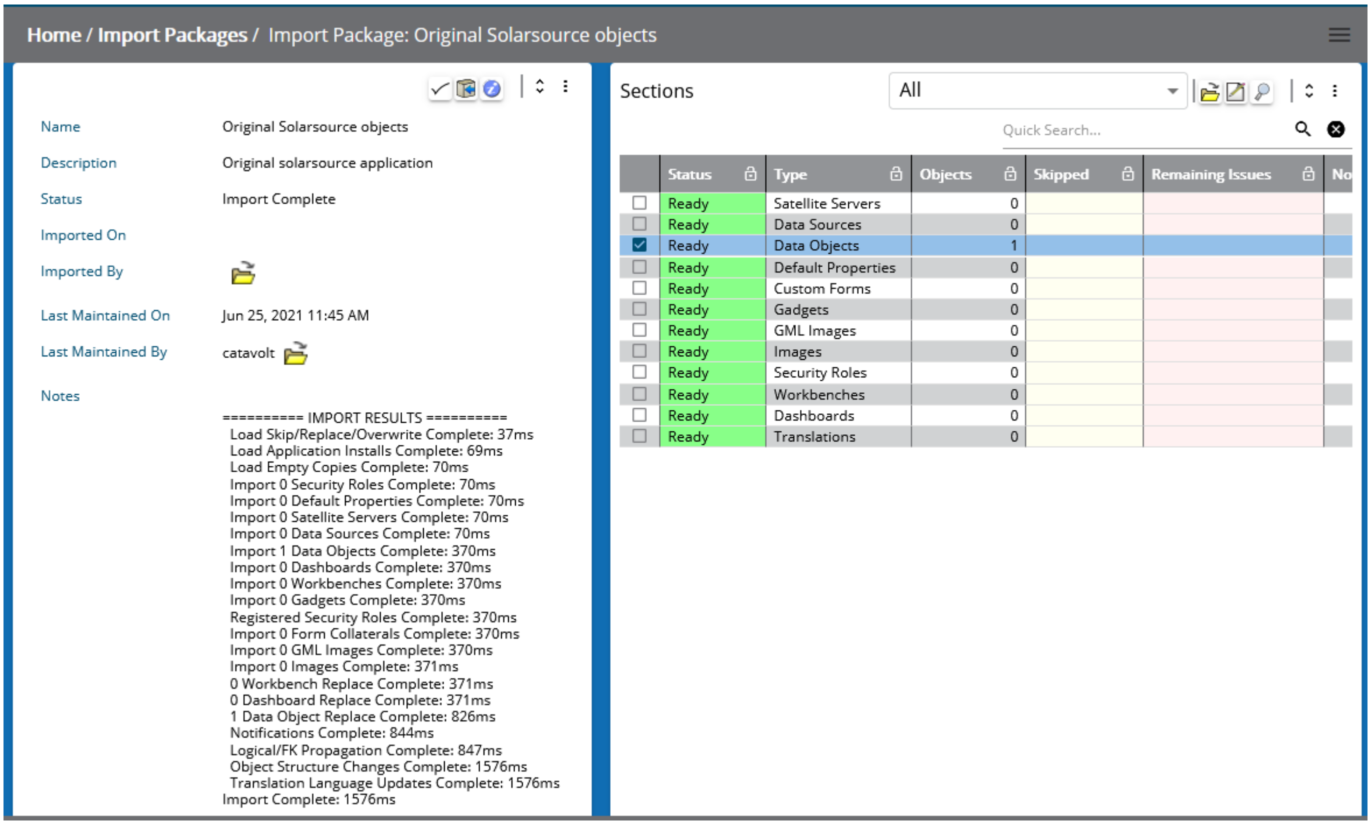
We recommend that you stay logged in to this session until after the Import has completed.

Figure 27: The Import Package Objects submission prompt

The Import Package's Status will change to Import In Progress while the package is being imported in the background. You are free to navigate around the application while the Import Package is being processed, although you should be cautious to not modify objects that the import process will overwrite. We also recommend that you remain logged in to your session until the Import is completed, although the Import will continue to completion even if you log off or are disconnected due to inactivity.

While the import is processing, you can run the "Check Import Status" action (see below) to see ongoing diagnostic information to ensure the package is still being processed in the background.

Once the import has successfully completed, the Import Package's Status will change to Import Complete. An IMPORT RESULTS section will be added to the end of Notes that contains the same diagnostic timings that are shown in the Check Import Status window.



Home / Import Packages / Import Package: Original Solarsource objects

Name	Original Solarsource objects
Description	Original solarsource application
Status	Import Complete
Imported On	
Imported By	
Last Maintained On	Jun 25, 2021 11:45 AM
Last Maintained By	catavolt
Notes	

```
========== IMPORT RESULTS ==========
  Load Skip/Replace/Overwrite Complete: 37ms
  Load Application Installs Complete: 69ms
  Load Empty Copies Complete: 70ms
  Import 0 Security Roles Complete: 70ms
  Import 0 Default Properties Complete: 70ms
  Import 0 Satellite Servers Complete: 70ms
  Import 0 Data Sources Complete: 70ms
  Import 1 Data Objects Complete: 370ms
  Import 0 Dashboards Complete: 370ms
  Import 0 Workbenches Complete: 370ms
  Import 0 Gadgets Complete: 370ms
  Registered Security Roles Complete: 370ms
  Import 0 Form Collaterals Complete: 370ms
  Import 0 GML Images Complete: 370ms
  Import 0 Images Complete: 371ms
  0 Workbench Replace Complete: 371ms
  0 Dashboard Replace Complete: 371ms
  1 Data Object Replace Complete: 826ms
  Notifications Complete: 844ms
  Logical/FK Propagation Complete: 847ms
  Object Structure Changes Complete: 1576ms
  Translation Language Updates Complete: 1576ms
Import Complete: 1576ms
```

Sections

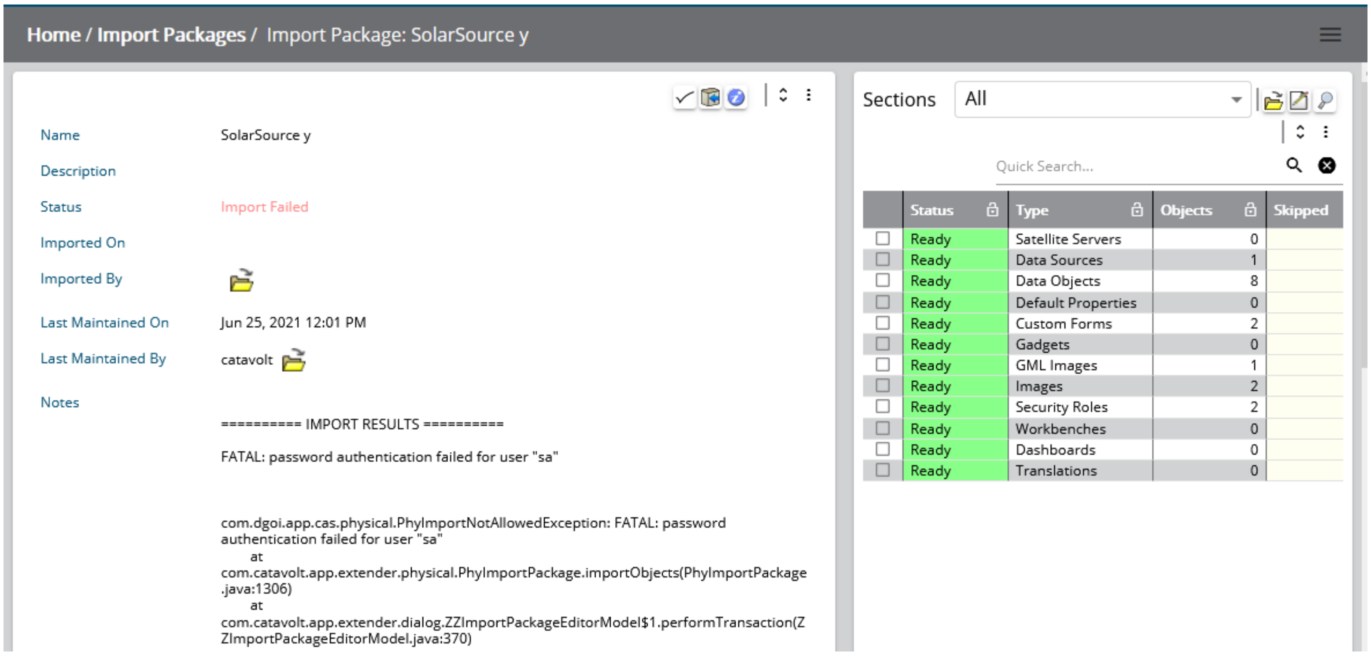
All

Quick Search...

	Status	Type	Objects	Skipped	Remaining Issues	No
□	Ready	Satellite Servers	0			
□	Ready	Data Sources	0			
☑	Ready	Data Objects	1			
□	Ready	Default Properties	0			
□	Ready	Custom Forms	0			
□	Ready	Gadgets	0			
□	Ready	GML Images	0			
□	Ready	Images	0			
□	Ready	Security Roles	0			
□	Ready	Workbenches	0			
□	Ready	Dashboards	0			
□	Ready	Translations	0			

Figure 28: The Import Package details after successful import.

If the Import fails for any reason, the Status will be changed to "Import Failed". An IMPORT RESULTS section will be added to the end of Notes that contains the full error information that occurred. You can resolve the noted error and retry the Import Objects action after a failed import.



Home / Import Packages / Import Package: SolarSource y

Name	SolarSource y
Description	
Status	Import Failed
Imported On	
Imported By	
Last Maintained On	Jun 25, 2021 12:01 PM
Last Maintained By	catavolt
Notes	

```
========== IMPORT RESULTS ==========

FATAL: password authentication failed for user "sa"


com.dgoi.app.cas.physical.PhyImportNotAllowedException: FATAL: password
authentication failed for user "sa"
        at
com.catavolt.app.extender.physical.PhyImportPackage.importObjects(PhyImportPackage
.java:1306)
        at
com.catavolt.app.extender.dialog.ZZImportPackageEditorModel$1.performTransaction(Z
ZImportPackageEditorModel.java:370)
```

Sections: All

Quick Search...

	Status	Type	Objects	Skipped
□	Ready	Satellite Servers	0	
□	Ready	Data Sources	1	
□	Ready	Data Objects	8	
□	Ready	Default Properties	0	
□	Ready	Custom Forms	2	
□	Ready	Gadgets	0	
□	Ready	GML Images	1	
□	Ready	Images	2	
□	Ready	Security Roles	2	
□	Ready	Workbenches	0	
□	Ready	Dashboards	0	
□	Ready	Translations	0	

Figure 29: The Import Package details after failed import.

You are free to delete the Import Package once the import has successfully completed. Import Packages cannot be reused, as the act of importing objects will fundamentally affect the environment and list of issues that need to be corrected.

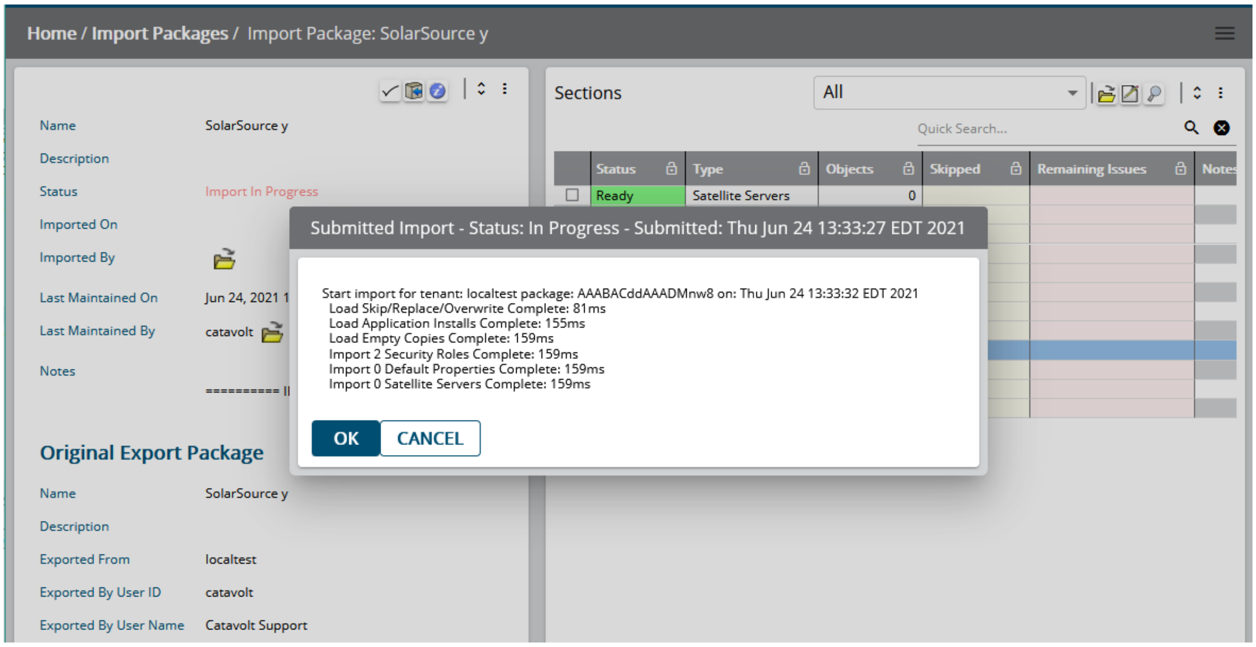
Other Import Menu Options

Recheck Import Package

The Recheck Import Package menu option allows you to do an automatic recheck of all the objects in the selected Import Package. In general, each time you resolve an Issue, the entire Import Package is rechecked, taking any resolutions into account. Some issues, however cannot be resolved from the Import Package and must be done outside the system (for example, going to the back-end database and adding a missing table). In cases where you have resolved an issue outside of the system, you can subsequently run the Recheck Import Package menu option to remove the issue from the Import Package.

Check Import Status

The Check Import Status menu option allows you to view the current status of an import being processed in the background. Running this action will bring up current diagnostic progress with timings to verify the Import is continuing:



Home / Import Packages / Import Package: SolarSource y

Name	SolarSource y
Description	
Status	Import In Progress
Imported On	
Imported By	
Last Maintained On	Jun 24, 2021 1
Last Maintained By	catavolt
Notes	========== II

Original Export Package

Name	SolarSource y
Description	
Exported From	localtest
Exported By User ID	catavolt
Exported By User Name	Catavolt Support

Sections: All

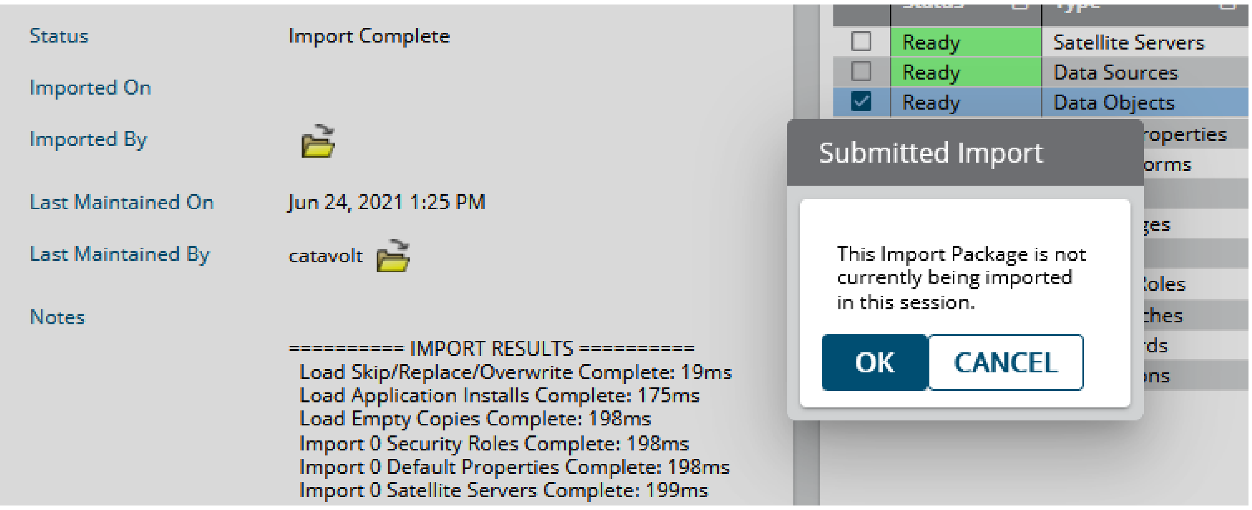
	Status	Type	Objects	Skipped	Remaining Issues	Notes
☐	Ready	Satellite Servers	0			

Submitted Import - Status: In Progress - Submitted: Thu Jun 24 13:33:27 EDT 2021

```
Start import for tenant: localtest package: AAABACddAAADMnw8 on: Thu Jun 24 13:33:32 EDT 2021
  Load Skip/Replace/Overwrite Complete: 81ms
  Load Application Installs Complete: 155ms
  Load Empty Copies Complete: 159ms
  Import 2 Security Roles Complete: 159ms
  Import 0 Default Properties Complete: 159ms
  Import 0 Satellite Servers Complete: 159ms
```

Figure 30: The Check Import Status window.

Choosing the action on an Import Package that is not being imported (or the import has already completed), or choosing the action from a different login session (i.e. not the session the import was initiated from) will show the following dialog:



Status Import Complete

Imported On

Imported By

Last Maintained On Jun 24, 2021 1:25 PM

Last Maintained By catavolt

Notes

```
========== IMPORT RESULTS ==========
  Load Skip/Replace/Overwrite Complete: 19ms
  Load Application Installs Complete: 175ms
  Load Empty Copies Complete: 198ms
  Import 0 Security Roles Complete: 198ms
  Import 0 Default Properties Complete: 198ms
  Import 0 Satellite Servers Complete: 199ms
```

□	Ready	Satellite Servers
□	Ready	Data Sources
☑	Ready	Data Objects

Submitted Import

This Import Package is not currently being imported in this session.

Figure 31: The Check Import Status window on a package not being imported.

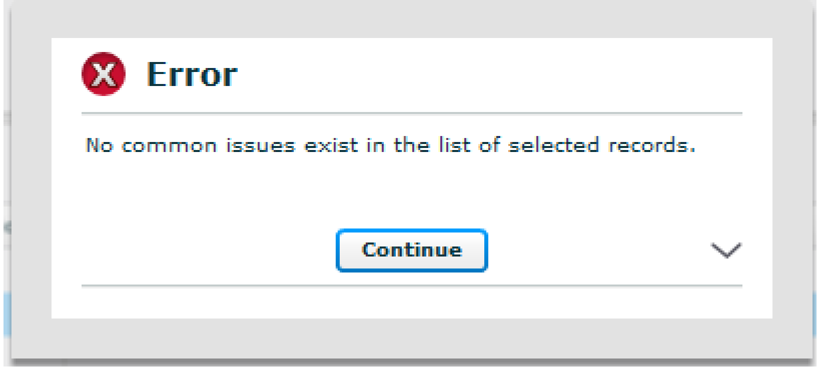
Resolve Common Issues

A common scenario during Export is to export an entire Data Source from a Development environment to a Production environment. The Data Source in Development is pointing to a Development database, while on Import you want the Data Source to point to a Production database. The proper way to handle this is to Skip Import on the Data Source itself. Instead, create a new Data Source pointing to the Production database and then go to each Data Object and resolve the "Data Object refers to Data Source 'xxx' that was not exported." issue with a Resolution of Reassign Reference that points to the new Data Source. Of course, if you have a lot of Data Objects, this can take some time. Another common scenario is to export data multiple times from one environment to another. On the second and subsequent imports, you see a number of 'This XXX has been previously imported into this environment.' issues. When importing large data sets, it can take a while to individually resolve these "expected" issues. In order to save time, we have added a multi-select Resolve Common Issues action on Import Section Objects. You can select multiple object and resolve issues all at once. When running this action, Xalt will scan through the selected records looking for common "expected" issues from the following list:

Object Type	Common Issues
Custom Forms	A custom form with the same name already exists in this environment.
Dashboard	This Dashboard has been previously imported into this environment.
Data Object	Data Object refers to Data Source 'XXX' that was not exported. This Data Object has been previously imported into this environment.

Data Source	Data Source refers to Connector gateway 'XXX' that was not exported.
Default Property	A Default Property with the same name already exists in this environment with a value of 'XXX'.
GML Images	An image with the same name already exists in this environment.
Images	An image with the same name already exists in this environment.
Security Role	A Security Role with the same name already exists in this environment.
Workbench	This Workbench has been previously imported into this environment.

If no selected records contain a common issue, the following error dialog is presented to the user:

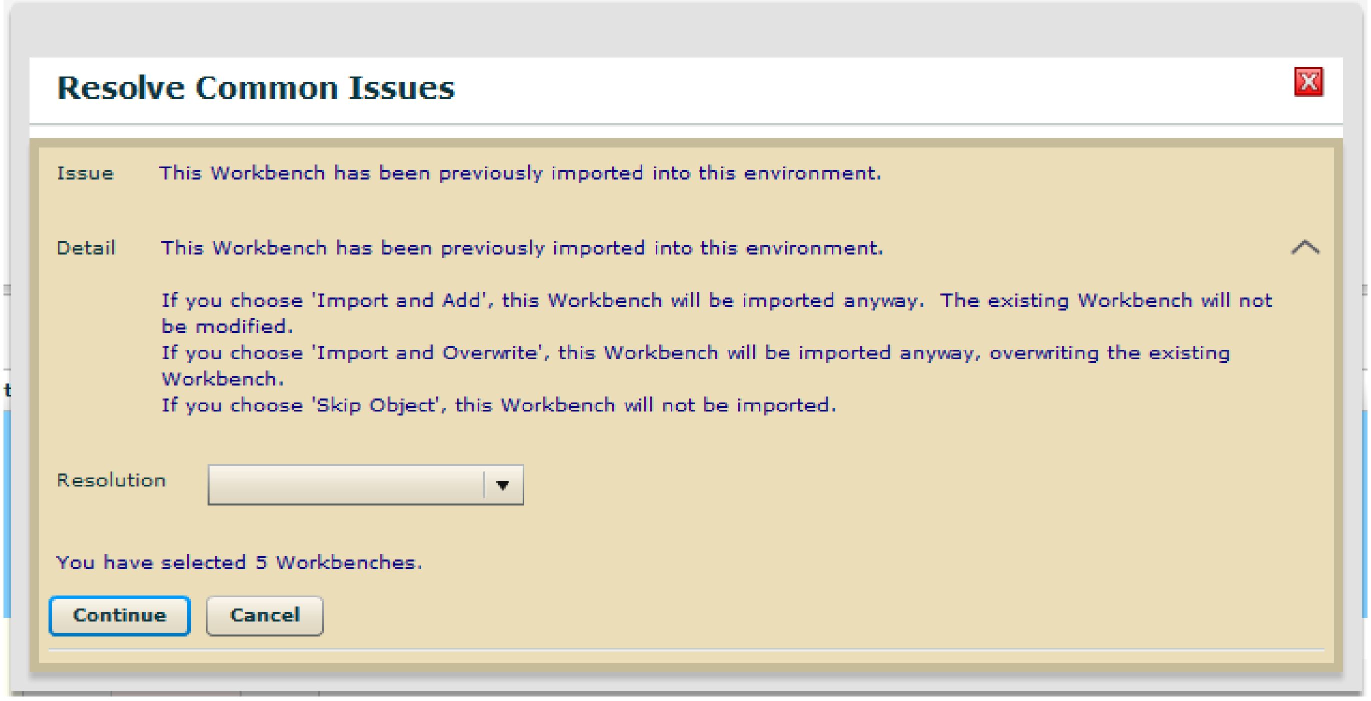


Error

No common issues exist in the list of selected records.

Figure 32: The Resolve Common Issues error prompt

If at least one object contains a common issue, you will be presented with a dialog containing the issue, detail on possible resolutions, and a list of Resolutions to choose from. This dialog is very similar to the window you see when manually resolving an issue.



Resolve Common Issues

Issue This Workbench has been previously imported into this environment.

Detail This Workbench has been previously imported into this environment.

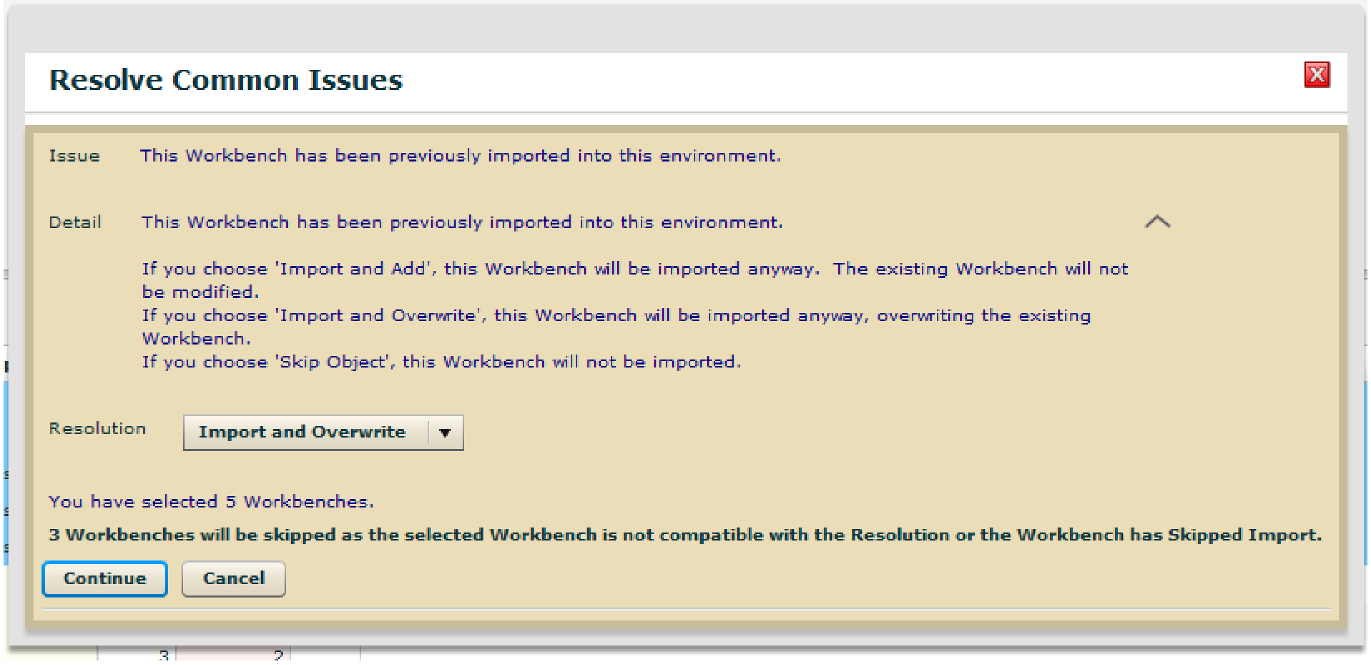
If you choose 'Import and Add', this Workbench will be imported anyway. The existing Workbench will not be modified.
If you choose 'Import and Overwrite', this Workbench will be imported anyway, overwriting the existing Workbench.
If you choose 'Skip Object', this Workbench will not be imported.

Resolution

You have selected 5 Workbenches.

Figure 33: The Resolve Common Issues prompt when run against Workbenches

Once you select a Resolution, a pre-check will be performed against the selected records. Since any invalid records will be skipped when the action is performed, the dialog will let you know the number of records that will be skipped before you run the action.



Resolve Common Issues

Issue This Workbench has been previously imported into this environment.

Detail This Workbench has been previously imported into this environment.

If you choose 'Import and Add', this Workbench will be imported anyway. The existing Workbench will not be modified.
If you choose 'Import and Overwrite', this Workbench will be imported anyway, overwriting the existing Workbench.
If you choose 'Skip Object', this Workbench will not be imported.

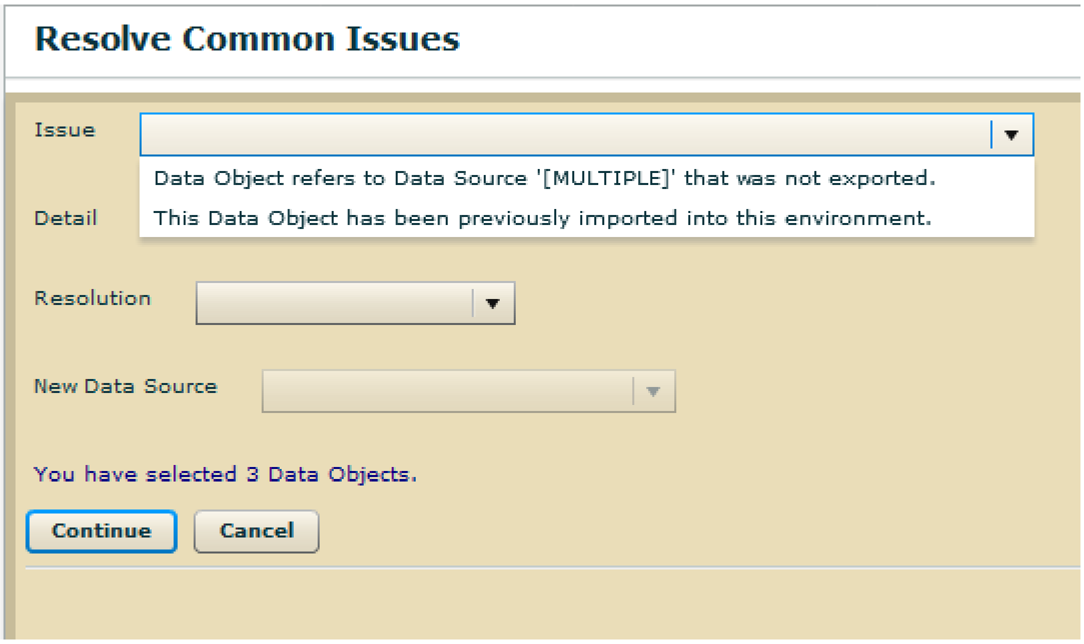
Resolution: Import and Overwrite

You have selected 5 Workbenches.

3 Workbenches will be skipped as the selected Workbench is not compatible with the Resolution or the Workbench has Skipped Import.

Figure 34: The Resolve Common Issues prompt after selecting a Resolution

A similar dialog is available when running Resolve Common Issues against a list of Data Objects, since Data Objects have multiple potential common issues. Selected records are scanned, and the Issue list will be filled with the superset of common issues from the selected records:



Resolve Common Issues

Issue

Data Object refers to Data Source '[MULTIPLE]' that was not exported.

This Data Object has been previously imported into this environment.

Detail

Resolution

New Data Source

You have selected 3 Data Objects.

Figure 35: The Resolve Common Issues prompt when run against Data Objects

Note that since these two issues are related (i.e. the 'This Data Object has been previously imported…' issue only appears for a Data Object that has previously resolved the 'Data Object refers to Data Source XXX that was not exported.' issue with a Reassign Reference resolution), you may find that you select a group of Data Objects and run Resolve Common Issues to find one issue available, then after resolving run Resolve Common Issues again to find the second issue now available.

The Resolve Common Issues action is meant to be a convenience method to resolve issues for multiple objects at once. It is functionally equivalent to going to each individual object and manually resolving the common issue. Because of this, you are able to run this action multiple times on the same object. You can also manually resolve issues that you previously ran this action against, and vice versa.

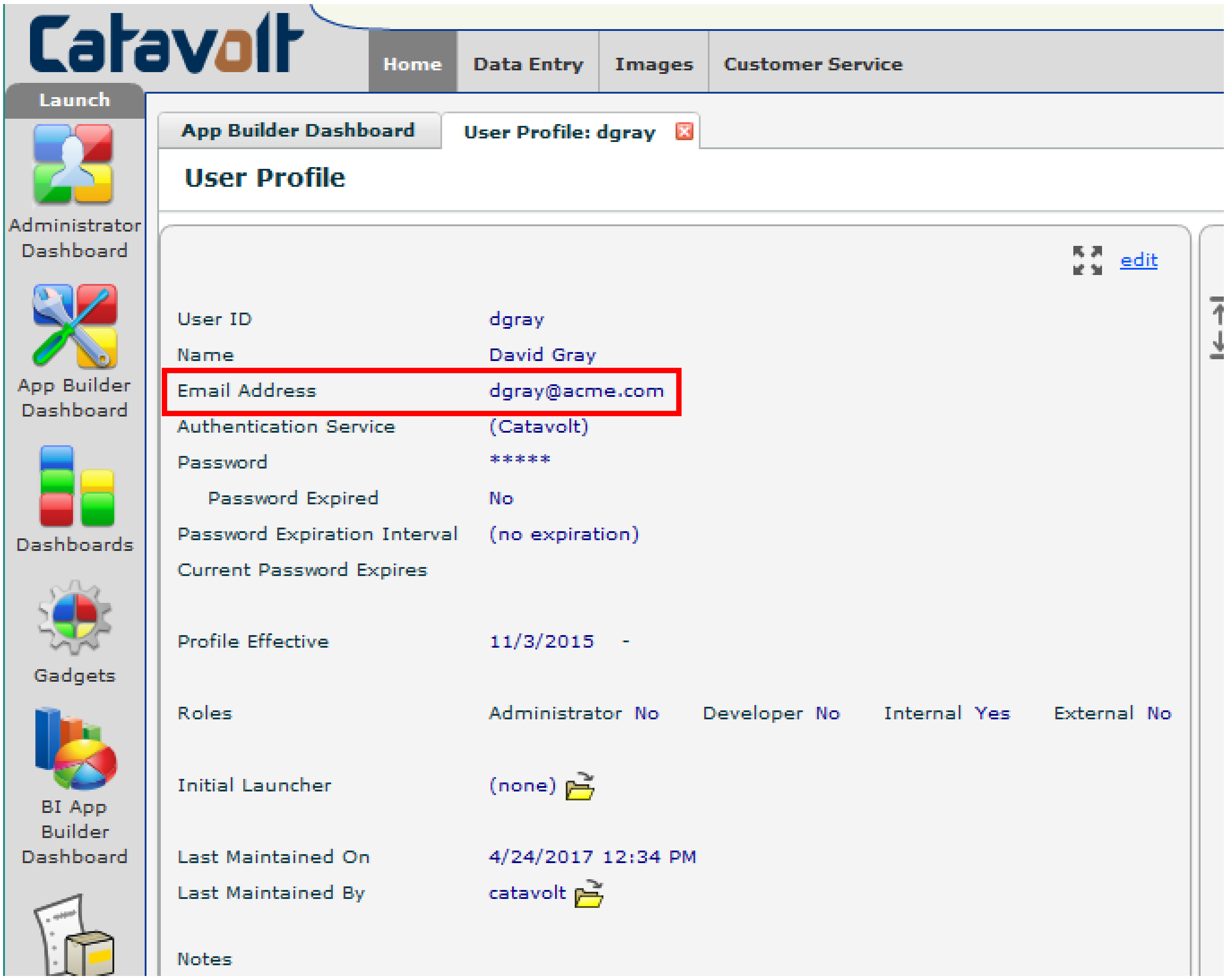
Finally, note that Data Object, Dashboard, and Workbench all have an 'This XXX has been previously imported into this environment.' issue. When manually resolving this issue, if you choose the "Import and Overwrite" resolution, you are required to select the matching object to replace. Consequently, if you choose the "Import and Overwrite" resolution when running Resolve Common Issues, the resolution will only be applied if there is exactly one matching object to overwrite. If there are multiple candidate objects to overwrite, the issue will be skipped and you will need to manually resolve it.

Chapter 16: Tenant Email Server

16

Chapter Summary

Xalt | Mobility has added the ability to send both manual and automatic emails to User Profiles. Xalt | Mobility will send the email through your own email server. Xalt | Mobility uses the Email Address defined in the User Profile record as the recipient of the email.



User Profile

User ID	dgray
Name	David Gray
Email Address	dgray@acme.com
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	11/3/2015 -
Roles	Administrator No Developer No Internal Yes External No
Initial Launcher	(none)
Last Maintained On	4/24/2017 12:34 PM
Last Maintained By	catavolt
Notes	

Figure 1: The User Profile record highlighting the Email Address showing the recipient of any emails to be sent

The Tenant Email Server is not able to send emails to external users (e.g. AD authenticated external users that map to a "group" Hexagon User Profile).

Accessing the Tenant Email Server

The Tenant Email Server is accessed through a new launcher on the Admin Home workbench – Email Server.

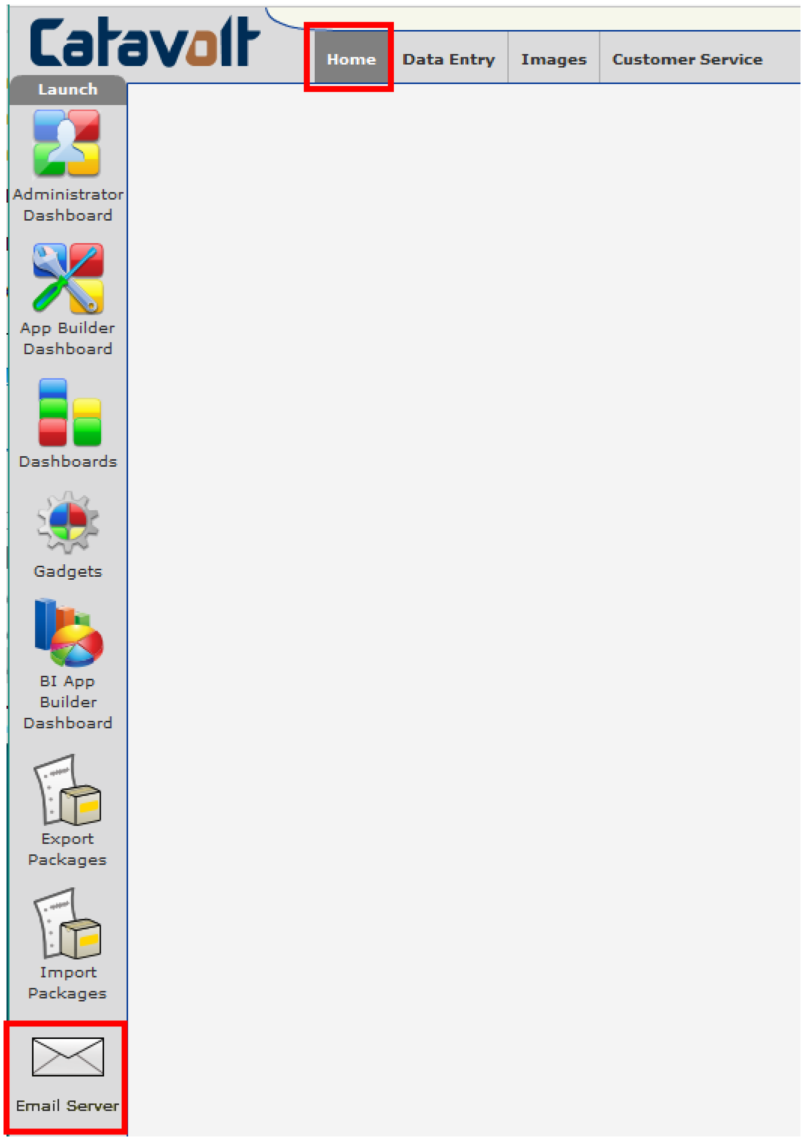
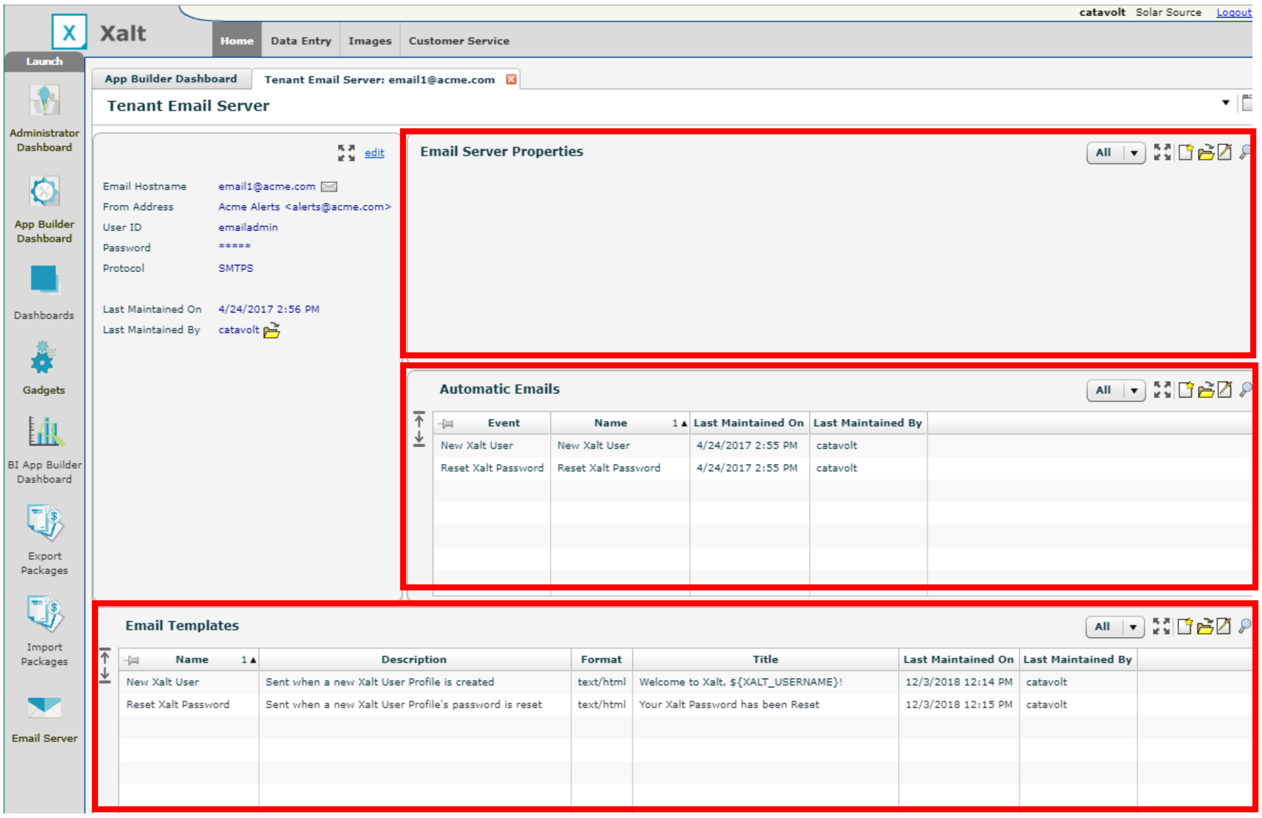


Figure 2: The Email Server launcher off the Admin Home Workbench

Tenant Email Server Components

There is only one Tenant Email Server object. The Tenant Email Server object has 3 sub-objects, "Email Server Properties", "Automatic Emails", and "Email Templates".



Tenant Email Server

Email Hostname	email1@acme.com
From Address	Acme Alerts <alerts@acme.com>
User ID	emailadmin
Password	*****
Protocol	SMTPS
Last Maintained On	4/24/2017 2:56 PM
Last Maintained By	catavolt

Email Server Properties

Automatic Emails

Event	Name	Last Maintained On	Last Maintained By
New Xalt User	New Xalt User	4/24/2017 2:55 PM	catavolt
Reset Xalt Password	Reset Xalt Password	4/24/2017 2:55 PM	catavolt

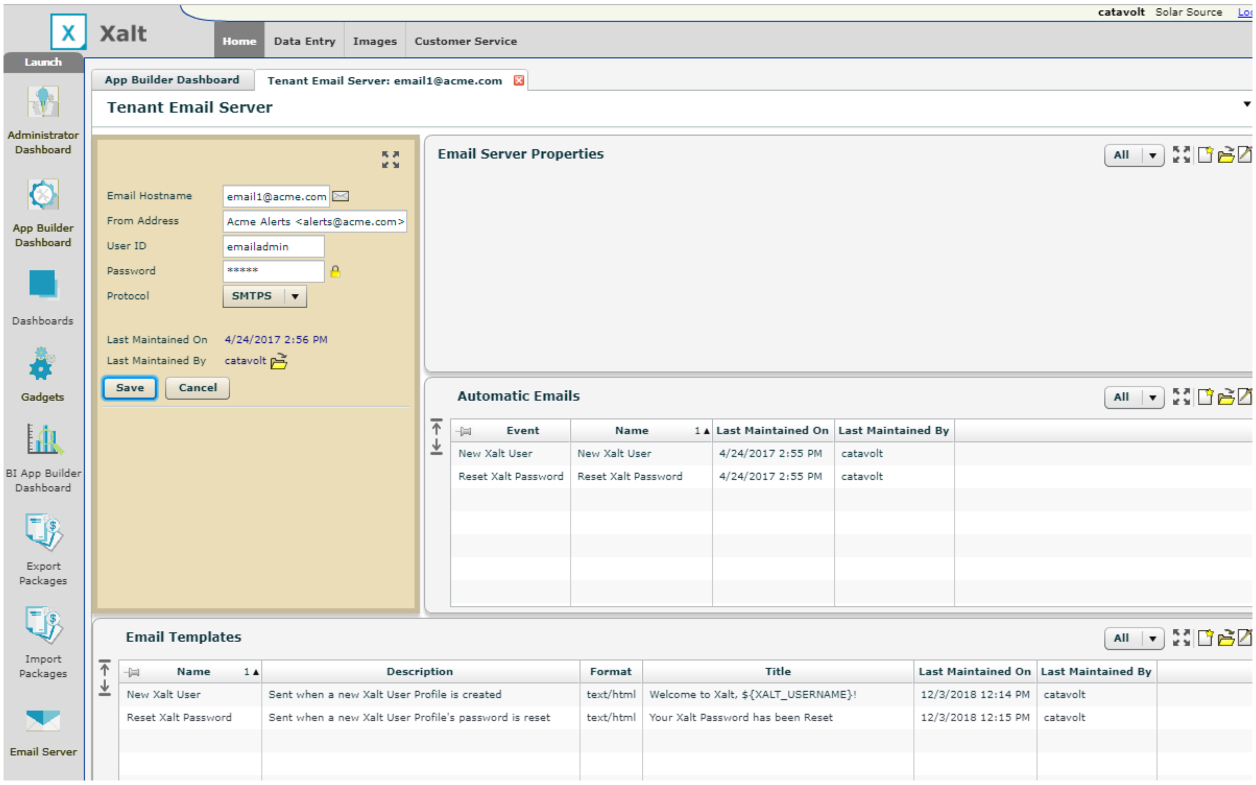
Email Templates

Name	Description	Format	Title	Last Maintained On	Last Maintained By
New Xalt User	Sent when a new Xalt User Profile is created	text/html	Welcome to Xalt, ${XALT_USERNAME}!	12/3/2018 12:14 PM	catavolt
Reset Xalt Password	Sent when a new Xalt User Profile's password is reset	text/html	Your Xalt Password has been Reset	12/3/2018 12:15 PM	catavolt

Figure 3: The Tenant Email Server details highlighting the Email Server Properties, Automatic Emails, and Email Templates sections.

Configuring the Tenant Email Server

By default, the Email Server is not configured. You must configure it using information from your email server.



Tenant Email Server

Email Hostname	email1@acme.com
From Address	Acme Alerts <alerts@acme.com>
User ID	emailadmin
Password	*****
Protocol	SMTPS
Last Maintained On	4/24/2017 2:56 PM
Last Maintained By	catavolt

Save Cancel

Automatic Emails

Event	Name 1▲	Last Maintained On	Last Maintained By
New Xalt User	New Xalt User	4/24/2017 2:55 PM	catavolt
Reset Xalt Password	Reset Xalt Password	4/24/2017 2:55 PM	catavolt

Email Templates

Name 1▲	Description	Format	Title	Last Maintained On	Last Maintained By
New Xalt User	Sent when a new Xalt User Profile is created	text/html	Welcome to Xalt, ${XALT_USERNAME}!	12/3/2018 12:14 PM	catavolt
Reset Xalt Password	Sent when a new Xalt User Profile's password is reset	text/html	Your Xalt Password has been Reset	12/3/2018 12:15 PM	catavolt

Figure 4: The Tenant Email Server details in update mode

Email Hostname specifies the external hostname or IP address of your email server. The Xalt Cloud Server will contact this server in order to request it to send emails.

From Address specifies the From: value in emails that are sent (the RFC 822 "From" header field).

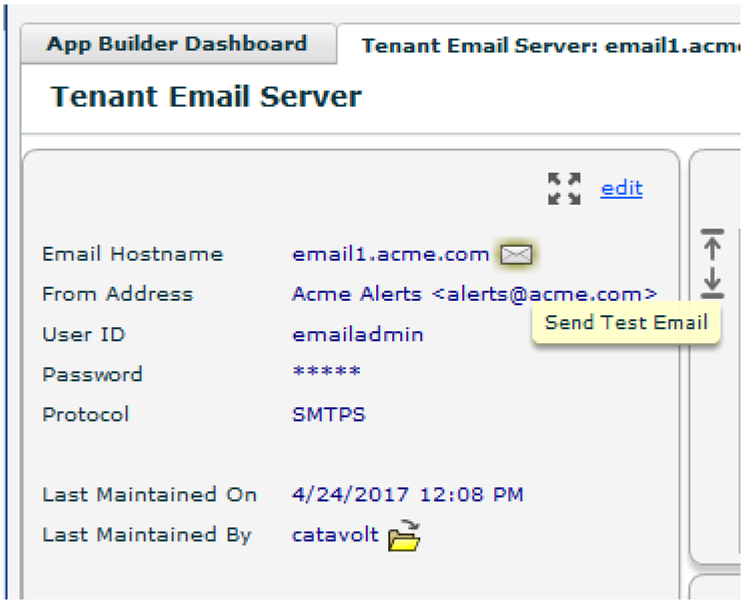
User ID specifies the User ID used to login to the email server.

Password specifies the Password used to login to the email server.

Protocol specifies the protocol to use to send emails. Allowable values are SMTPS (default) and SMTP.

Send Test Email

Once you have set up your Tenant Email Server, you can test it using the "Send Test Email" button next to the Email Hostname:

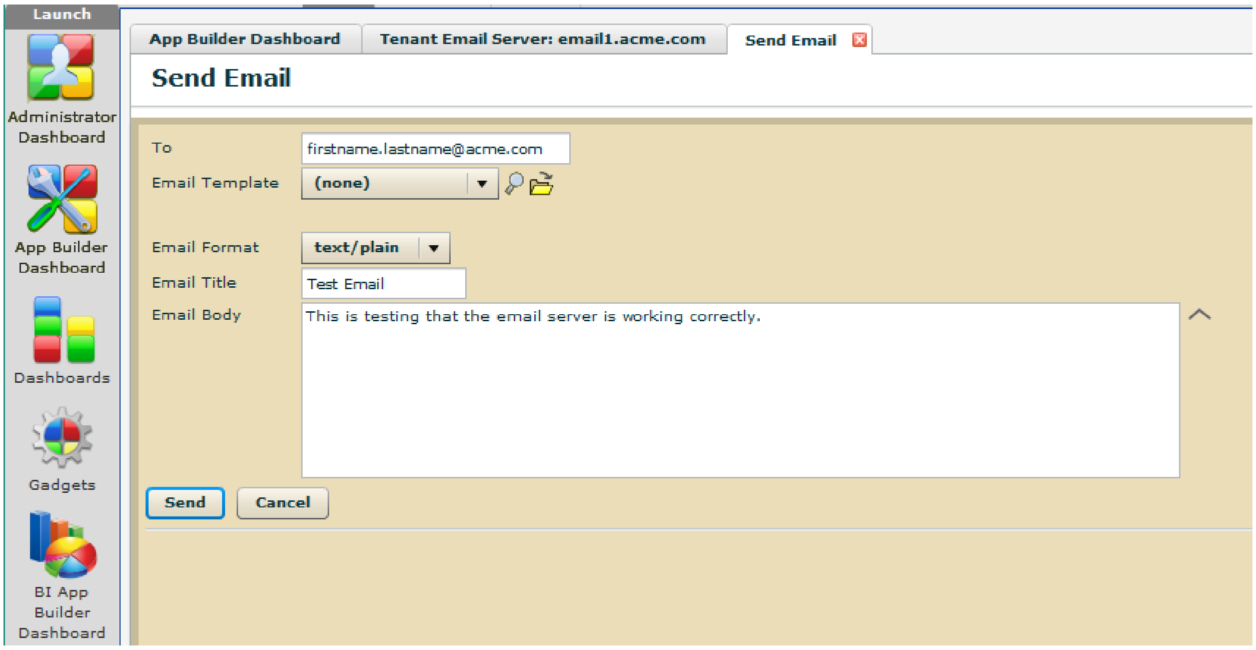


Tenant Email Server

Email Hostname	email1.acme.com
From Address	Acme Alerts <alerts@acme.com>
User ID	emailadmin
Password	*****
Protocol	SMTPS
Last Maintained On	4/24/2017 12:08 PM
Last Maintained By	catavolt

Figure 5: The Tenant Email Server details highlighting the Send Test Email action

Selecting this action will bring up the Send Email window:



Send Email

To	firstname.lastname@acme.com
Email Template	(none)
Email Format	text/plain
Email Title	Test Email
Email Body	This is testing that the email server is working correctly.

Figure 6: The Send Email action window

To specifies the email address to receive the email

Email Template allow you to (optionally) choose an existing Email Template as a starting point. Selecting a value will automatically fill in the Email Format, Email Title, and Email Body fields from the selected Email Template.

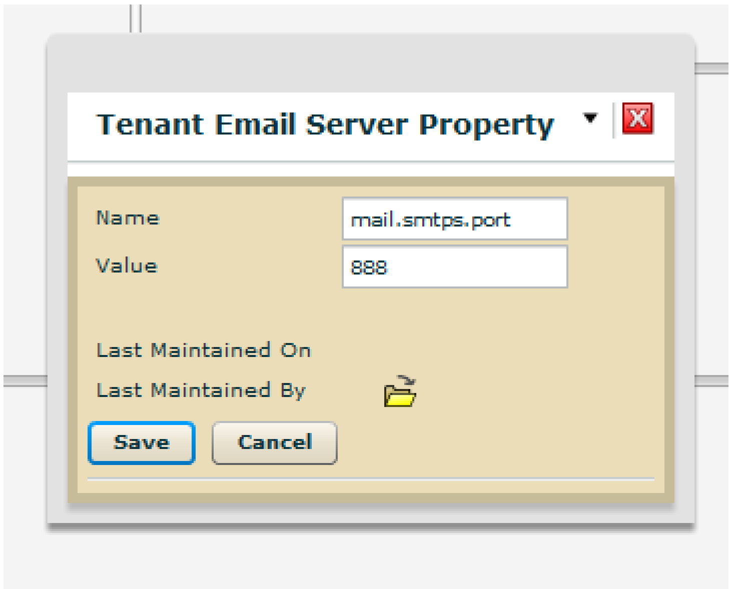
Email Format specifies the format of the email to be sent. Allowable values are text/plain (default) and text/html.

Email Title specifies the title of the email.

Email Body specifies the body of the email.

Email Server Properties

Email Server Properties specifies any extra properties that you need passed to your email server during connection (e.g. if your email server is running on a non-standard port). This section will be blank for most standard email installations.



Tenant Email Server Property

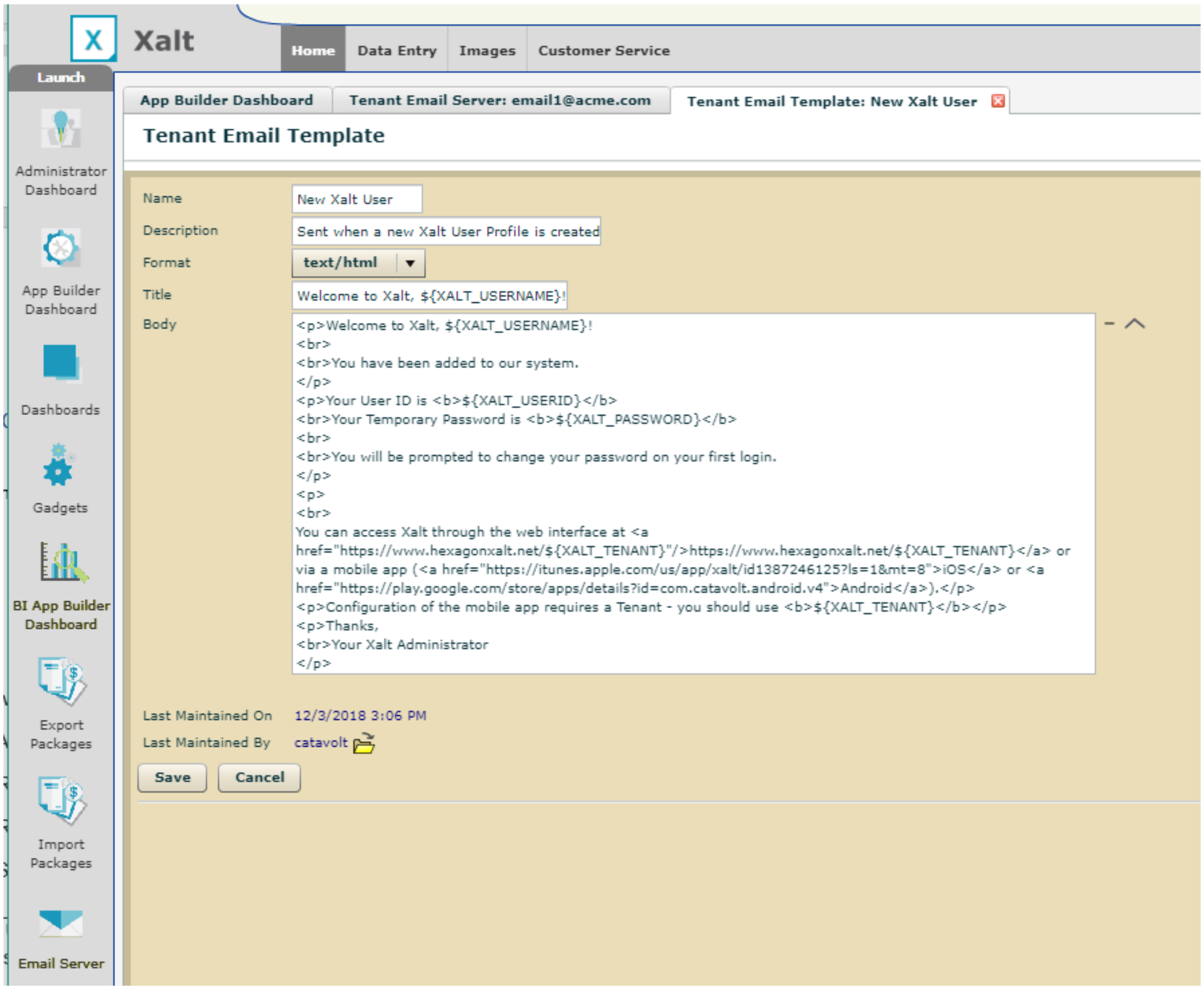
Name	mail.smtps.port
Value	888
Last Maintained On	
Last Maintained By	

Figure 7: The Tenant Email Server Property details in update mode

Enter the Property Name and Value. All properties will be passed to the email server on connection.

Email Templates

Email Templates allow you to create reusable emails that can be sent multiple times. Xalt ships with some sample Email Templates that you are free to use, alter, or delete. You can also create new Email Templates:



Tenant Email Template

Name: New Xalt User

Description: Sent when a new Xalt User Profile is created

Format: text/html

Title: Welcome to Xalt, ${XALT_USERNAME}!

Body:

```
<p>Welcome to Xalt, ${XALT_USERNAME}!
<br>
<br>You have been added to our system.
</p>
<p>Your User ID is <b>${XALT_USERID}</b>
<br>Your Temporary Password is <b>${XALT_PASSWORD}</b>
<br>
<br>You will be prompted to change your password on your first login.
</p>
<p>
<br>
You can access Xalt through the web interface at <a
href="https://www.hexagonxalt.net/${XALT_TENANT}"/>https://www.hexagonxalt.net/${XALT_TENANT}</a> or
via a mobile app (<a href="https://itunes.apple.com/us/app/xalt/id1387246125?ls=1&mt=8">iOS</a> or <a
href="https://play.google.com/store/apps/details?id=com.catavolt.android.v4">Android</a>).</p>
<p>Configuration of the mobile app requires a Tenant - you should use <b>${XALT_TENANT}</b></p>
<p>Thanks,
<br>Your Xalt Administrator
</p>
```

Last Maintained On: 12/3/2018 3:06 PM

Last Maintained By: catavolt

Figure 8: The Tenant Email Template details in update mode

Name specifies the name of the Email Template.

Description specifies the description of the Email Template.

Format specifies the format of the email to be sent. Allowable values are text/plain (default) and text/html.

Title specifies the title of the email. This value can contain Substitution Values (see below).

Body specifies the body of the email. This value can contain Substitution Values (see below).

Title and Body have support for Substitution Values. Xalt will use the following Substitution Value types:

User Properties (per each User Profile)

Default Properties

Standard Properties

In addition, the following special values are allowed for Email Templates

XALT_TENANT – Your tenant ID (e.g. acme)

XALT_USERID – The User ID property from the specified User Profile

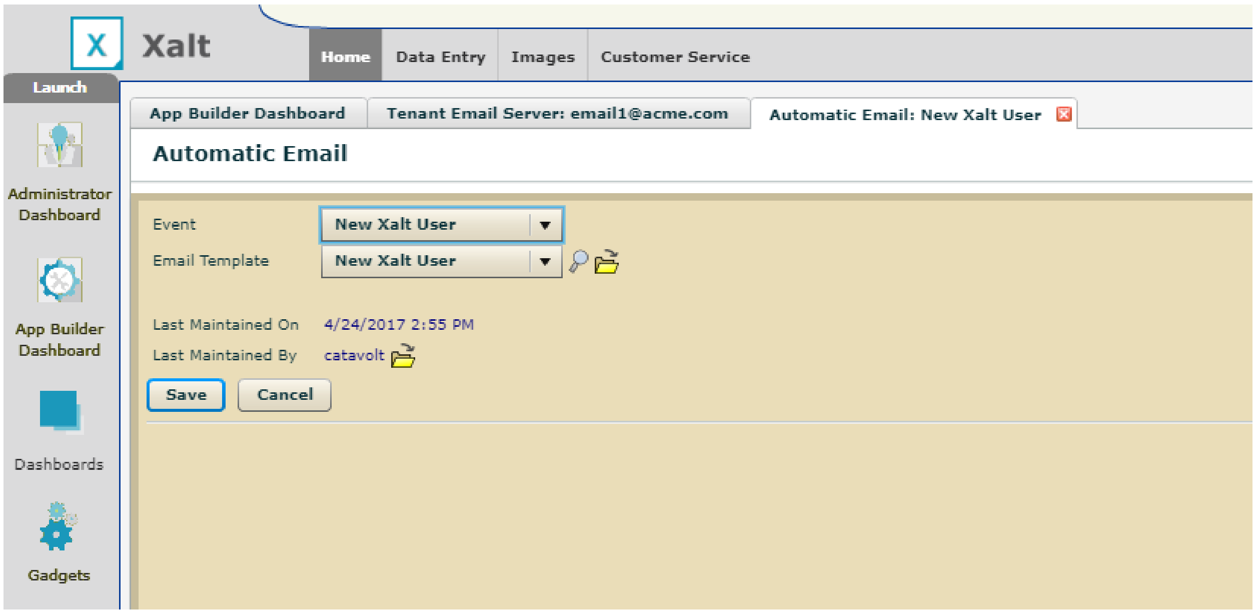
XALT_USERNAME – The Name property from the specified User Profile

XALT_PASSWORD – The Password property from the specified User Profile

Note that the XALT_PASSWORD property will only be available when sending an Automatic Email for "New Xalt User" or "Reset Xalt Password". This Substitution Value will return a blank when used in other situations.

Automatic Emails

You can specify Automatic Emails to be sent to users when certain events occur.



Automatic Email

Event: New Xalt User

Email Template: New Xalt User

Last Maintained On: 4/24/2017 2:55 PM

Last Maintained By: catavolt

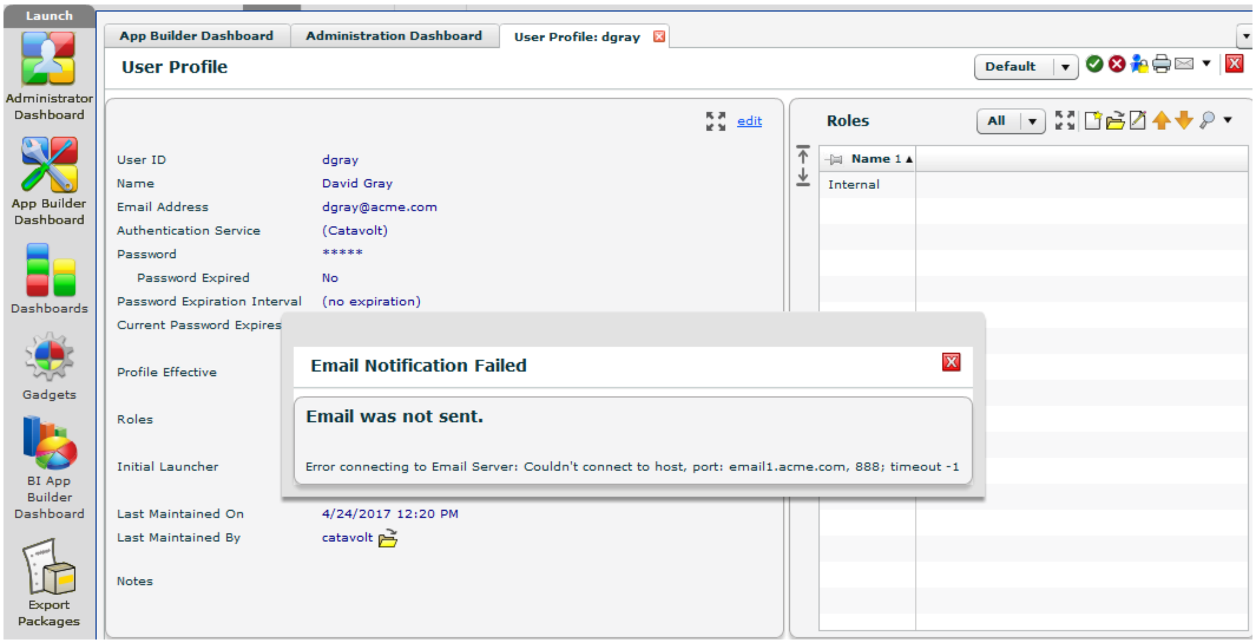
Figure 9: The Automatic Email details in update mode

Event specifies the type of event. Allowable values are:

- New Xalt User – A new User Profile with Xalt Authentication has been created
- Reset Xalt Password – An administrator has changed the password on a User Profile with Xalt Authentication

Email Template specifies the Email Template to send when the event occurs. Any Substitution Values will be processed using values from the User Profile being changed.

When you perform an event that has an Automatic Email, the email will be silently sent. You will receive an error window if the email server could not send the email or if Xalt could not connect to the Email Server:



User Profile

User ID dgray
Name David Gray
Email Address dgray@acme.com
Authentication Service (Catavolt)
Password *****
Password Expired No
Password Expiration Interval (no expiration)
Current Password Expires
Profile Effective
Roles
Initial Launcher
Last Maintained On 4/24/2017 12:20 PM
Last Maintained By catavolt
Notes

Internal

Email Notification Failed

Email was not sent.

Error connecting to Email Server: Couldn't connect to host, port: email1.acme.com, 888; timeout -1

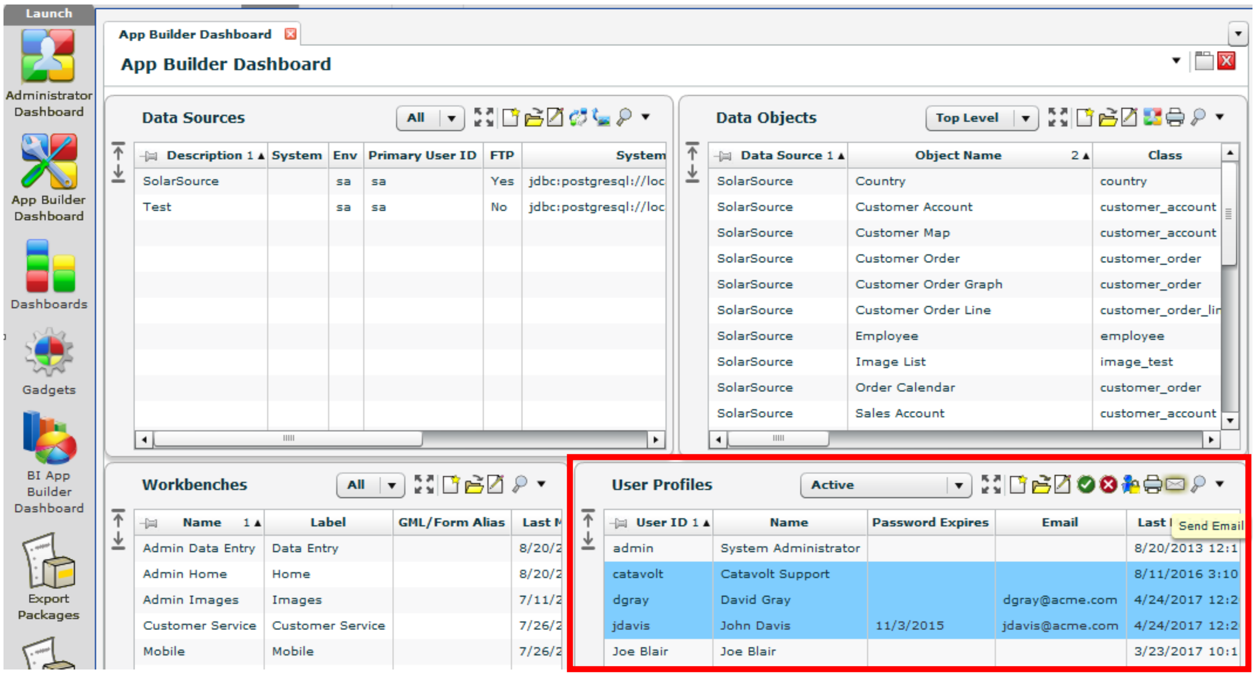
Figure 10: An error window when an automatic email failed to send

Manual Emails

You can also manually send emails to Xalt User Profiles from a number of locations.

User Profiles

You can select one or more User Profile records and choose the Send Email action:



Data Sources

Description 1 ▲	System	Env	Primary User ID	FTP	System
SolarSource		sa	sa	Yes	jdbc:postgresql://loc
Test		sa	sa	No	jdbc:postgresql://loc

Data Objects

Data Source 1 ▲	Object Name 2 ▲	Class
SolarSource	Country	country
SolarSource	Customer Account	customer_account
SolarSource	Customer Map	customer_account
SolarSource	Customer Order	customer_order
SolarSource	Customer Order Graph	customer_order
SolarSource	Customer Order Line	customer_order_lir
SolarSource	Employee	employee
SolarSource	Image List	image_test
SolarSource	Order Calendar	customer_order
SolarSource	Sales Account	customer_account

Workbenches

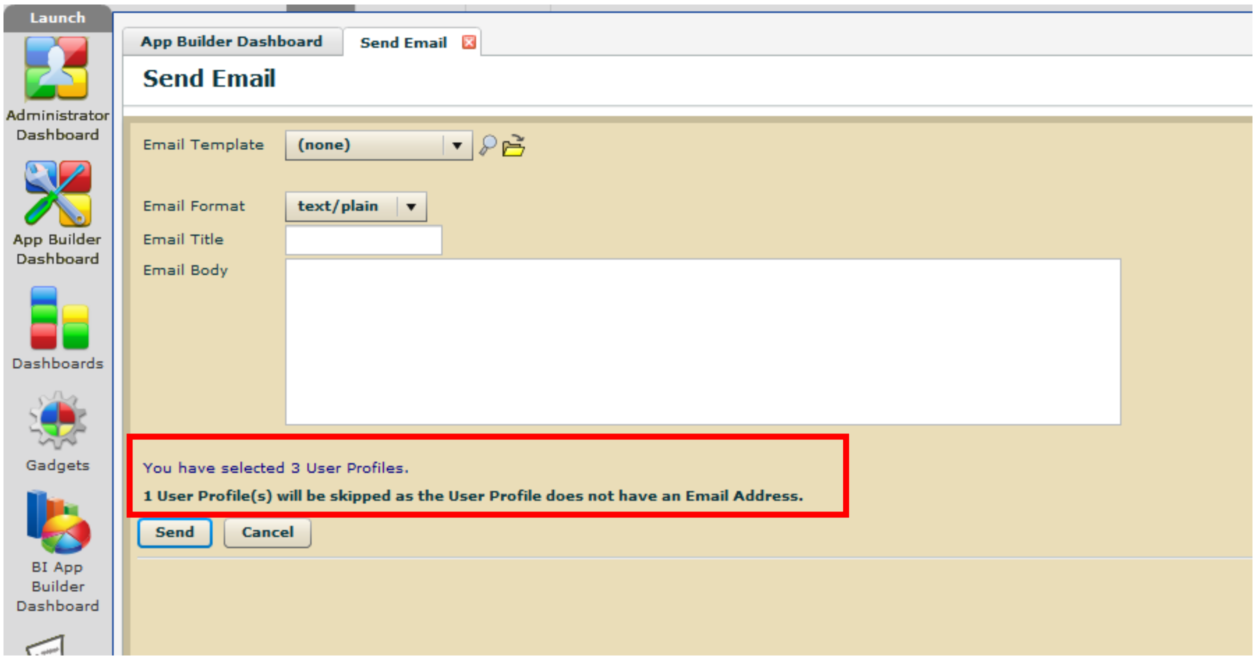
Name 1 ▲	Label	GML/Form Alias	Last M
Admin Data Entry	Data Entry		8/20/2
Admin Home	Home		8/20/2
Admin Images	Images		7/11/2
Customer Service	Customer Service		7/26/2
Mobile	Mobile		7/26/2

User Profiles

User ID 1 ▲	Name	Password Expires	Email	Last
admin	System Administrator			8/20/2013 12:1
catavolt	Catavolt Support			8/11/2016 3:10
dgray	David Gray		dgray@acme.com	4/24/2017 12:2
jdavis	John Davis	11/3/2015	jdavis@acme.com	4/24/2017 12:2
Joe Blair	Joe Blair			3/23/2017 10:1

Figure 11: The User Profiles list highlighting the Send Email action

Choosing this option will bring up the following window:



Send Email

Email Template (none)

Email Format text/plain

Email Title

Email Body

You have selected 3 User Profiles.

1 User Profile(s) will be skipped as the User Profile does not have an Email Address.

Figure 12: The Send Email action window highlighting the number of users selected and skipped

Email Template allows you to (optionally) choose an existing Email Template as a starting point. Selecting a value will automatically fill in the Email Format, Email Title, and Email Body fields from the selected Email Template.

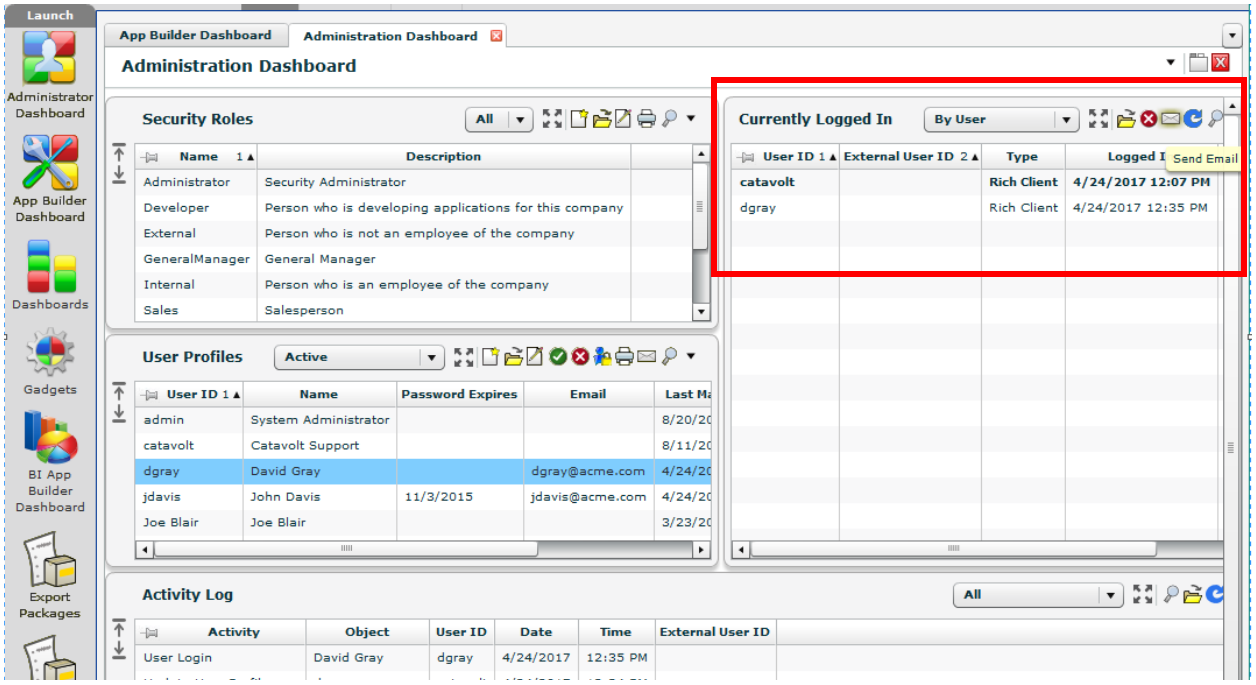
Email Format specified the format of the email to be sent. Allowable values are text/plain (default) and text/html.

Email Title specifies the title of the email.

Email Body specifies the body of the email.

The window will tell you the number of User Profiles selected as well as the number of User Profiles that will be skipped because they do not contain an Email Address. The email will be sent to the valid selected User Profiles. Substitution Values such as HEXAGON_USERID, User Properties, etc. will be run separately for each user so that each user's email will contain their own specific values.

The same Send Email button is available from the Currently Logged In section of the Administration Dashboard:



Administration Dashboard

Security Roles

Name 1 ▲	Description
Administrator	Security Administrator
Developer	Person who is developing applications for this company
External	Person who is not an employee of the company
GeneralManager	General Manager
Internal	Person who is an employee of the company
Sales	Salesperson

Currently Logged In

User ID 1 ▲	External User ID 2 ▲	Type	Logged I
catavolt		Rich Client	4/24/2017 12:07 PM
dgray		Rich Client	4/24/2017 12:35 PM

User Profiles

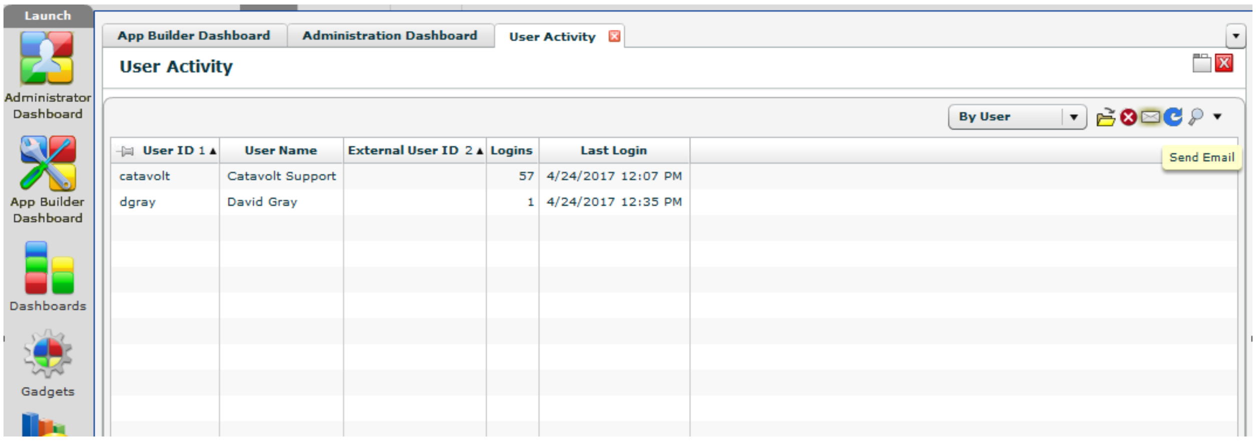
User ID 1 ▲	Name	Password Expires	Email	Last Ma
admin	System Administrator			8/20/20
catavolt	Catavolt Support			8/11/20
dgray	David Gray		dgray@acme.com	4/24/20
jdavis	John Davis	11/3/2015	jdavis@acme.com	4/24/20
Joe Blair	Joe Blair			3/23/20

Activity Log

Activity	Object	User ID	Date	Time	External User ID
User Login	David Gray	dgray	4/24/2017	12:35 PM	

Figure 13: The Currently Logged In window highlighting the Send Email action

As well as the Show User Activity list:

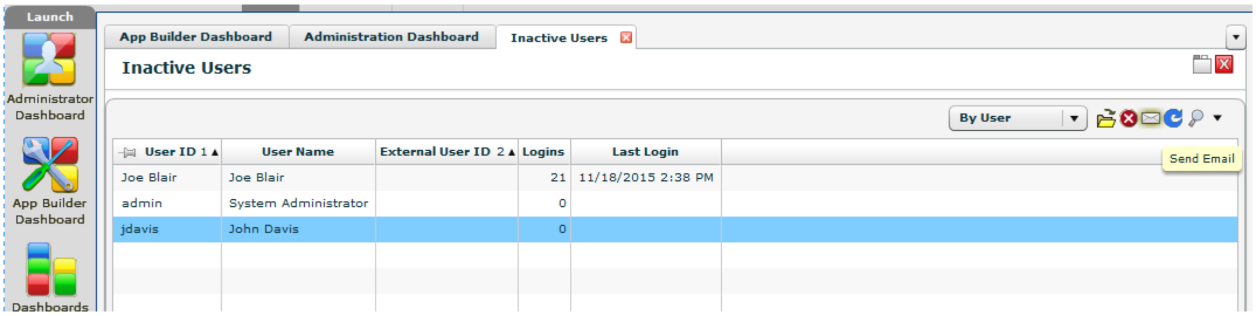


User Activity

User ID 1 ▲	User Name	External User ID 2 ▲	Logins	Last Login
catavolt	Catavolt Support		57	4/24/2017 12:07 PM
dgray	David Gray		1	4/24/2017 12:35 PM

Figure 14: The User Activity window highlighting the Send Email action

And Show Inactive Users list:



Inactive Users

User ID 1 ▲	User Name	External User ID 2 ▲	Logins	Last Login
Joe Blair	Joe Blair		21	11/18/2015 2:38 PM
admin	System Administrator		0	
jdavis	John Davis		0	

Figure 15: The Inactive Users window highlighting the Send Email action

Appendices

Appendix A: Specifying Messages and Substitution Values

There are a number of places in Xalt where you can provide a message to be displayed to the end user. Some examples are the Short and Descriptive Title fields on Data Object, the Left and Right Operand fields on Data Annotation Rule, etc. A message can consist of constant values (such as 'Purchase Orders' or 51) combined with Substitution Values. A Substitution Value is a placeholder for dynamic information that is calculated at runtime. Substitution Values are surrounded by ${} and are case-sensitive. There are 7 basic types of Substitution values.

Standard Substitution Values

Standard Substitution Values are values that are available to all users across the system and are based on standard values. Dates used in substitution values are based on the standard calendar year and not fiscal year. The following is a list of available Standard Substitution Values:

CURRENT_DATE	Today
CURRENT_MONTH_END	The last day of the current month
CURRENT_MONTH_START	The first day of the current month
CURRENT_QUARTER_END	The last day of the current quarter
CURRENT_QUARTER_START	The first day of the current quarter
CURRENT_TIME	Now (Time)
CURRENT_TIMESTAMP	Now (Date + Time)
CURRENT_USER	The user ID currently logged into Hexagon
CURRENT_USER_EMAIL	The email for the user currently logged into Hexagon
CURRENT_USER_NAME	The name for the user currently logged into Hexagon
CURRENT_USER_ORGANIZATION	The Organization for the user currently logged into Hexagon
CURRENT_USER_ORGANIZATION_DESCRIPTION	The Organization description for the user currently logged into Hexagon
CURRENT_USER_ORGANIZATION_LEVEL	The Organization Level for the user currently logged into Hexagon. Will be "PREMIUM" for Organization Premium Users, "BASIC" for other members of an Organization, or blank if the user does not have an Organization defined.
CURRENT_USER_ORGANIZATION_ROLE	The Organization role for the user currently logged into Hexagon. Will be "ADMIN" for Organization Administrators, "MEMBER" for members of an Organization, or blank if the user does not have an Organization defined.
CURRENT_USER_PHONE	The phone for the user currently logged into Hexagon
CURRENT_EFFECTIVE_USER	Either CURRENT_USER or

	CURRENT_EXTERNAL_USER_ID, depending on whether the user is internal (Hexagon) or external (assigned via Custom Login Provider).
CURRENT_EXTERNAL_USER_ID	The user ID currently logged into Hexagon, if using the Authenticate External Users option on a Data Source
CURRENT_CLIENT_DATE	Current date on the client device (browser or mobile device)
CURRENT_CLIENT_DEVICEID	Notification ID of the mobile device
CURRENT_CLIENT_DEVICE_NAME	Basic Description of mobile device
CURRENT_CLIENT_PLATFORM	WEB (rich browser), IOS (iPhone/iPad), or ANDROID (Android device)
CURRENT_CLIENT_TIME	Current time on the client device (browser or mobile device)
CURRENT_CLIENT_TIMESTAMP	Current date + time on the client device (browser or mobile device)
CURRENT_CLIENT_TIMEZONE	Current time zone on the client device (browser or mobile device)
CURRENT_CLIENT_VERSION	Product version on the client device (browser or mobile device)
CURRENT_WEEK_END	The last day of the current week
CURRENT_WEEK_START	The first day of the current week
CURRENT_YEAR_END	The last day of the current year
CURRENT_YEAR_START	The first day of the current year
LAST_MONTH_END	The last day of the previous month
LAST_MONTH_START	The first day of the previous month
LAST_QUARTER_END	The last day of the previous quarter
LAST_QUARTER_START	The first day of the previous quarter
LAST_WEEK_END	The last day of the previous week
LAST_WEEK_START	The first day of the previous week
LAST_YEAR_END	The last day of the previous year
LAST_YEAR_START	The first day of the previous year
NEXT_MONTH_END	The last day of the following month
NEXT_MONTH_START	The first day of the following month
NEXT_QUARTER_END	The last day of the following quarter
NEXT_QUARTER_START	The first day of the following quarter
NEXT_WEEK_END	The last day of the following week
NEXT_WEEK_START	The first day of the following week
NEXT_YEAR_END	The last day of the following year
NEXT_YEAR_START	The first day of the following year

Date values, Time values, Timestamp values, and User values also support functions to allow you to specify these values in other formats as well as generate related dynamic values.

Date values (CURRENT_DATE, NEXT_MONTH_END, etc) support the following syntax:

<DATE FUNCTION>+|-nnnnDAY(S)|WEEK(S)|MONTH(S)|YEAR(S)

For example:

```
${CURRENT_DATE+5DAYS}
${CURRENT_MONTH_END-10WEEKS}
${LAST_YEAR_START-1YEAR}
```

Time values (CURRENT_TIME, CURRENT_CLIENT_TIME) support the following syntax:

```
<TIME FUNCTION>+|-nnnnHOUR(S) |MINUTE(S) |SECOND(S)
```

For example:

```
${CURRENT_TIME+1HOUR}
${CURRENT_TIME-10MINUTES}
${CURRENT_TIME+125SECONDS}
```

Timestamp values (CURRENT_TIMESTAMP, CURRENT_CLIENT_TIMESTAMP) support the following syntax:

```
<TIMESTAMP FUNCTION>+|-nnnn DAY(S)|WEEK(S)|MONTH(S)|YEAR(S)|HOUR(S) |MINUTE(S) |SECOND(S)
```

For example:

```
${CURRENT_TIMESTAMP+125DAYS}
${CURRENT_TIMESTAMP+1HOUR}
${CURRENT_TIMESTAMP-10MINUTES}
```

Date, Time, Timestamp, and User functions also support specific formatting options to display the value.

Date functions support the following syntax:

```
<DATE FUNCTION>@CYYMMDD|YYYYMMDD|DATE|DATETIME|CENTURY|YEAR|QUARTER_OF_YEAR|MONTH_OF_YEAR|
WEEK_OF_YEAR|DAY_OF_MONTH|DAY_OF_WEEK|DAY_OF_YEAR
```

For example:

```
${CURRENT_DATE@YYYYMMDD}
```

The formats are:

- CYYMMDD – A 7-digit value (C=Century (20th=0, 21st=1,etc), YY=2 digit Year, MM=Month, DD=Day)
- YYYYMMDD – An 8-digit value (YYYY=4 digit Year, MM=Month, DD=Day)
- DATE – A String in the format YYYY-MM-DD
- DATETIME – A String in the format YYYY-MM-DDTHH:MM:SS (Note: The time portion will be set to 00:00:00)
- CENTURY – A numeric value specifying the century (e.g. 20 for 2017)
- YEAR – A numeric value specifying the year (e.g. 2017)
- QUARTER_OF_YEAR – A numeric value between 1 and 4 specifying the calendar quarter
- MONTH_OF_YEAR – A numeric value between 1 and 12 specifying the calendar month
- WEEK_OF_YEAR – A numeric value between 1 and 53 specifying the ISO 8601 week number
- DAY_OF_MONTH – A numeric value between 1 and 12 specifying the calendar month

- DAY_OF_WEEK – A numeric value between 1 (Monday) and 7 (Sunday) specifying the calendar day of the week
- DAY_OF_YEAR – A numeric value between 1 and 366 specifying the calendar day of the year

Time functions support the following syntax:

<TIME FUNCTION>@HHMMSS|HHMM|TIME|DATETIME|HOUR_OF_DAY| MINUTE_OF_HOUR|SECOND_OF_MINUTE

For example:

${CURRENT_TIME@HHMMSS}

The formats are:

- HHMMSS – A 6-digit value (HH=Hour (24-hour format), MM=Minutes, SS=Seconds)
- HHMM – A 4-digit value (HH=Hour (24-hour format), MM=Minutes)
- TIME – A String in the format HH:MM:SS
- DATETIME – A String in the format YYYY-MM-DDTHH:MM:SS (Note: The date portion will be set to 1970-01-01)
- HOUR_OF_DAY – A numeric value between 0 and 23 specifying the hour
- MINUTE_OF_HOUR – A numeric value between 0 and 59 specifying the minute
- SECOND_OF_MINUTE – A numeric value between 0 and 59 specifying the second

Timestamp functions support the following syntax:

<TIMESTAMP FUNCTION>@CYYMMDD|YYYYMMDD|DATE|DATETIME|CENTURY|YEAR|QUARTER_OF_YEAR|MONTH_OF_YEAR|WEEK_OF_YEAR|DAY_OF_MONTH|DAY_OF_WEEK|DAY_OF_YEAR|HHMMSS|HHMM|TIME|HOUR_OF_DAY|MINUTE_OF_HOUR|SECOND_OF_MINUTE

For example:

${CURRENT_TIMESTAMP@DATETIME}

The formats are:

- CYYMMDD – A 7-digit value (C=Century (20th=0, 21st=1,etc), YY=2 digit Year, MM=Month, DD=Day)
- YYYYMMDD – An 8-digit value (YYYY=4 digit Year, MM=Month, DD=Day)
- DATE – A String in the format YYYY-MM-DD
- DATETIME – A String in the format YYYY-MM-DDTHH:MM:SS
- CENTURY – A numeric value specifying the century (e.g. 20 for 2017)
- YEAR – A numeric value specifying the year (e.g. 2017)
- QUARTER_OF_YEAR – A numeric value between 1 and 4 specifying the calendar quarter
- MONTH_OF_YEAR – A numeric value between 1 and 12 specifying the calendar month
- WEEK_OF_YEAR – A numeric value between 1 and 53 specifying the ISO 8601 week number
- DAY_OF_MONTH – A numeric value between 1 and 12 specifying the calendar month
- DAY_OF_WEEK – A numeric value between 1 (Monday) and 7 (Sunday) specifying the calendar day of the week
- DAY_OF_YEAR – A numeric value between 1 and 366 specifying the calendar day of the year

- HHMMSS – A 6-digit value (HH=Hour (24-hour format), MM=Minutes, SS=Seconds)
- HHMM – A 4-digit value (HH=Hour (24-hour format), MM=Minutes)
- TIME – A String in the format HH:MM:SS
- HOUR_OF_DAY – A numeric value between 0 and 23 specifying the hour
- MINUTE_OF_HOUR – A numeric value between 0 and 59 specifying the minute
- SECOND_OF_MINUTE – A numeric value between 0 and 59 specifying the second

Note that in cases where you specify both a date function and formatting option, the formatting option MUST be last. For example: ${CURRENT_DATE+1WEEK@CYYMMDD}

CURRENT_USER, CURRENT_EFFECTIVE_USER, CURRENT_EXTERNAL_USER_ID, CURRENT_USER_EMAIL, CURRENT_USER_ORGANIZATION, CURRENT_USER_PHONE and CURRENT_USER_NAME support specific formatting options to display the value.

<CURRENT USER FUNCTION>@UPPERCASE|LOWERCASE

For example:

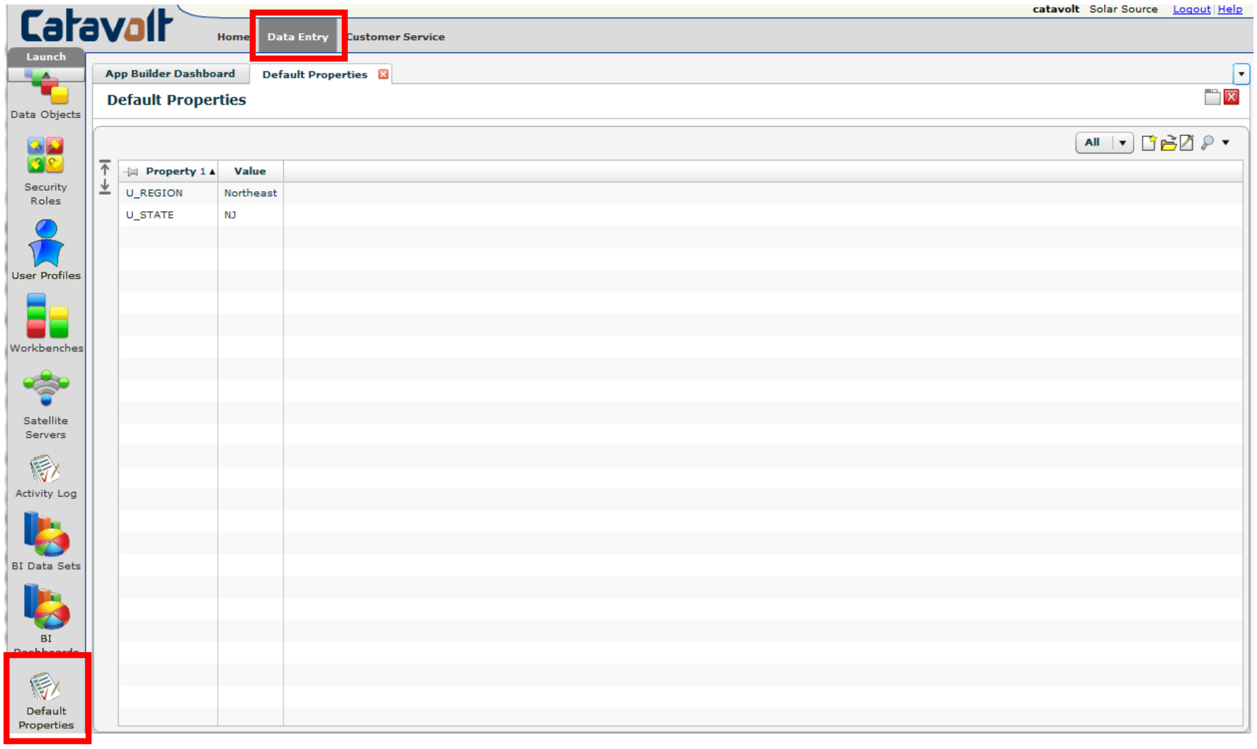
${CURRENT_USER@UPPERCASE}

The formats are:

- UPPERCASE– Display this mixed case value in all Uppercase (capital) letters
- LOWERCASE– Display this mixed case value in all Lowercase (non-capital) letters

Default Substitution Values

Default Substitution Values exist across the entire system. You can add them by going to the Default Properties launcher in the Data Entry Workbench.



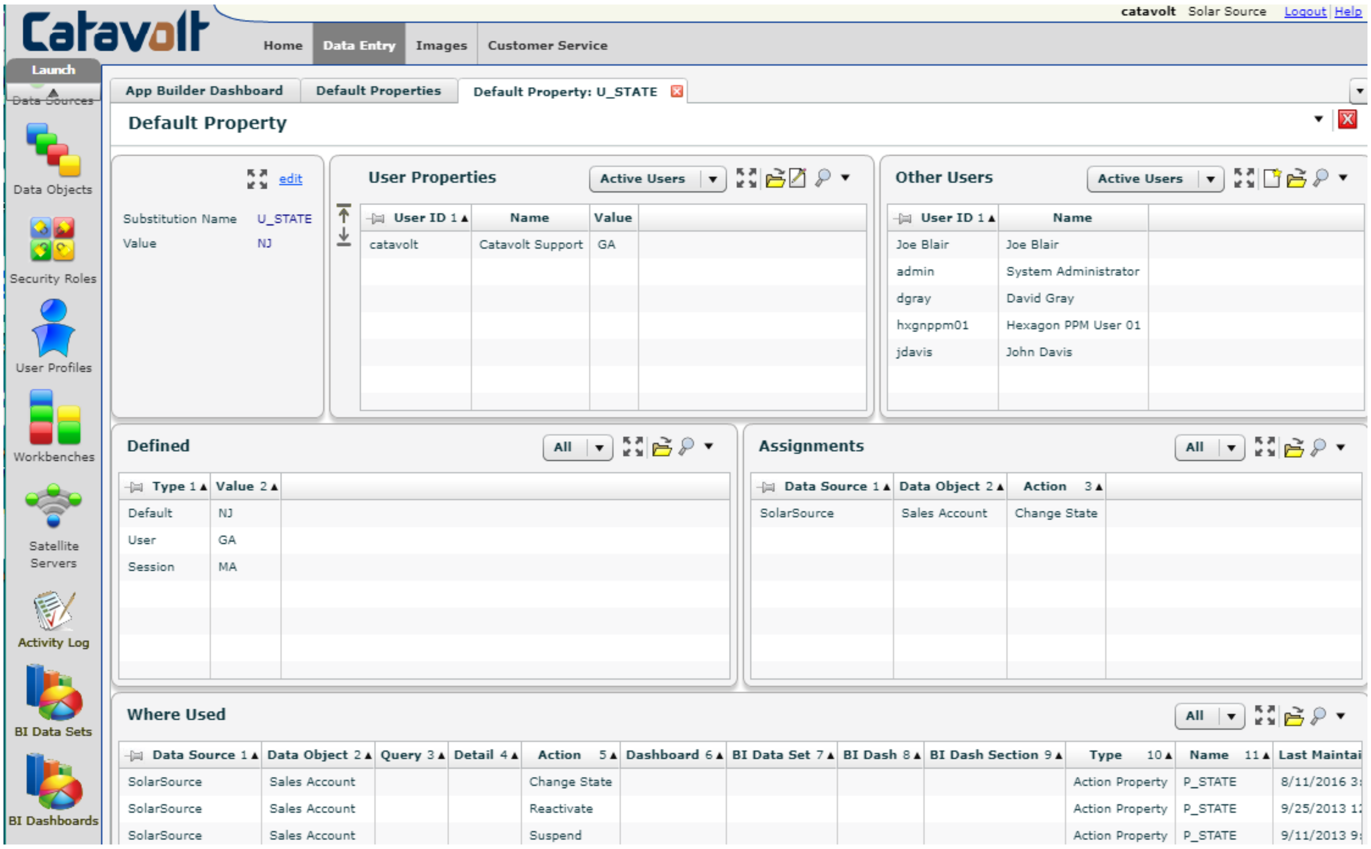
Default Properties

Property 1 ▲	Value
U_REGION	Northeast
U_STATE	NJ

Figure 1: Default Properties list with Workbench and Launcher highlighted

As these values are case-sensitive, it is typically a best practice to use all capitals when defining these names so as not to conflict with any Data Object property names. These values can be used any place where substitution values are allowed. Default Substitution Values are used to set a base value for a Substitution value that can then be overridden by creating User Substitution Values, Session Substitution Values, etc. that have the same Substitution Name.

Double-clicking a Default Property will display the following window:



Substitution Name	U_STATE
Value	NJ

User Properties

User ID 1 ▲	Name	Value
catavolt	Catavolt Support	GA

Other Users

User ID 1 ▲	Name
Joe Blair	Joe Blair
admin	System Administrator
dgray	David Gray
hxgnppm01	Hexagon PPM User 01
jdavis	John Davis

Defined

Type 1 ▲	Value 2 ▲
Default	NJ
User	GA
Session	MA

Assignments

Data Source 1 ▲	Data Object 2 ▲	Action 3 ▲
SolarSource	Sales Account	Change State

Where Used

Data Source 1 ▲	Data Object 2 ▲	Query 3 ▲	Detail 4 ▲	Action 5 ▲	Dashboard 6 ▲	BI Data Set 7 ▲	BI Dash 8 ▲	BI Dash Section 9 ▲	Type 10 ▲	Name 11 ▲	Last Maintai
SolarSource	Sales Account			Change State					Action Property	P_STATE	8/11/2016 3:
SolarSource	Sales Account			Reactivate					Action Property	P_STATE	9/25/2013 1:
SolarSource	Sales Account			Suspend					Action Property	P_STATE	9/11/2013 9:

Figure 2: Default Properties details view

The Details has the following sub-sections:

User Properties

User Properties shows all User Profiles that have a User Property defined over the same name as the Default Property. You can edit one or more records to change the value.

Other Users

Other Users shows all User Profiles that do not have a User Property defined over the same name as the Default Property. You can select one or more records to create a User Property for these User Profiles.

Defined

Defined shows the current levels where this property is defined. You may have a record defined at the Default (Default Property), User (User Profile Property) and/or Session (Session Property) levels. Double-clicking a Default record will bring up details for the Default Property. Double-clicking a User record will allow you to edit the User Profile Property value.

Assignments

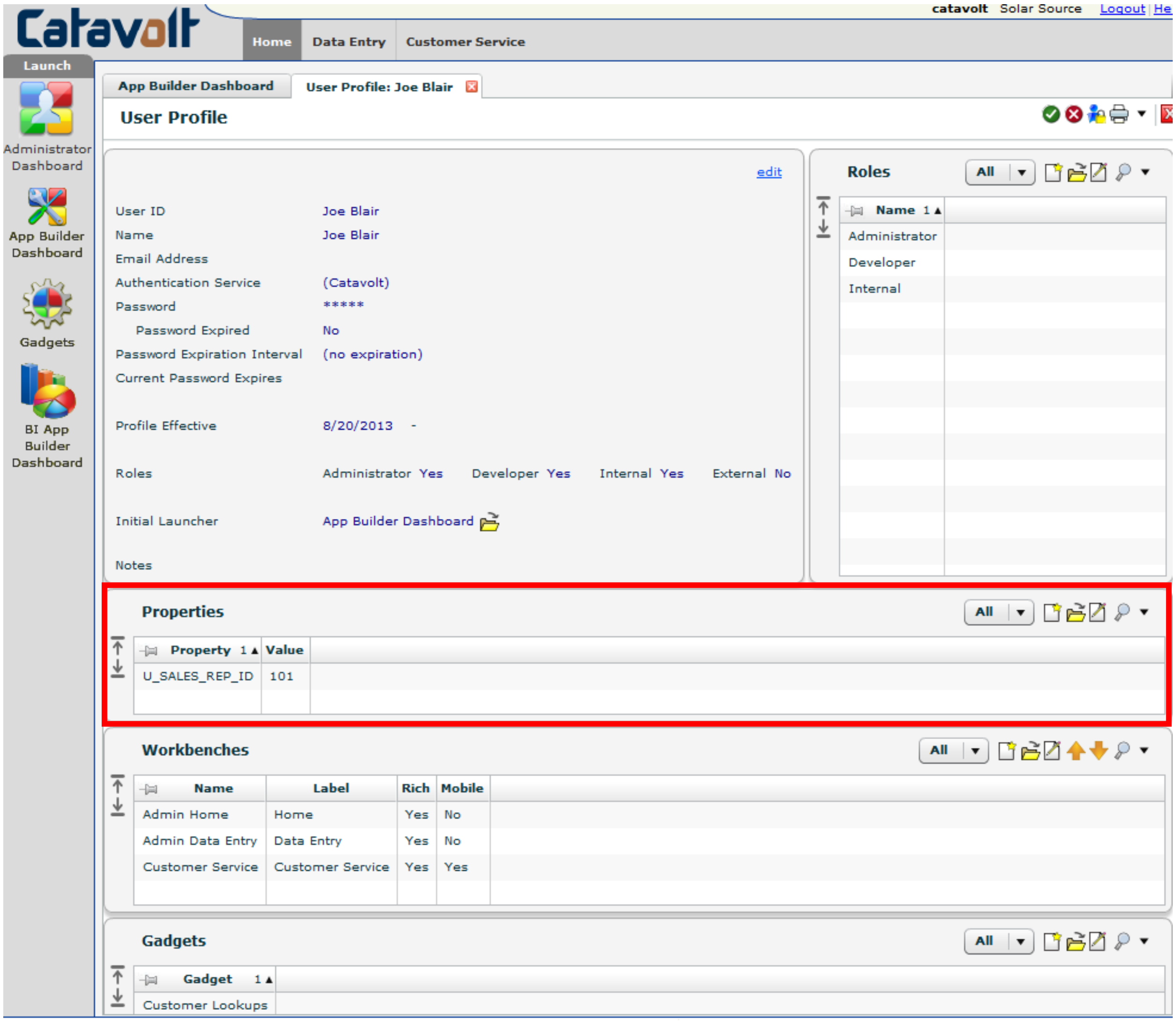
Assignments shows Action Request XML sections where this value is set via a SetSessionProperty, SetUserProperty, or SetDefaultProperty action. You can double-click the record to bring up the referenced Action.

Where Used

Where Used shows all of the objects that are currently using this property. You can double-click the record to bring up the referenced object.

User Substitution Values

User Substitution Values exist for a specific User Profile. You can add them by going to the Properties section of a User Profile. As these values are case-sensitive, it is typically a best practice to use all capitals when defining these names so as not to conflict with any Data Object property names. These values can be used any place where substitution values are allowed. However, if a User Property is used as a substitution value (e.g. on a where clause or Action XML request) and the currently logged in user does not have that property defined, an error will occur when performing the request. User Substitution Values are used in places where the data returned or actions performed need to be different depending on the current user. Typical uses are in where clauses to restrict access to certain records or in dashboard launchers to select a record specific to the current user. If a User Substitution Value and a Default Substitution Value have the same Substitution Name, the User Substitution Value will be used.



User Profile

User ID	Joe Blair
Name	Joe Blair
Email Address	
Authentication Service	(Catavolt)
Password	*****
Password Expired	No
Password Expiration Interval	(no expiration)
Current Password Expires	
Profile Effective	8/20/2013 -
Roles	Administrator Yes Developer Yes Internal Yes External No
Initial Launcher	App Builder Dashboard
Notes	

Roles

Name 1 ▲
Administrator
Developer
Internal

Properties

Property 1 ▲	Value
U_SALES_REP_ID	101

Workbenches

Name	Label	Rich	Mobile
Admin Home	Home	Yes	No
Admin Data Entry	Data Entry	Yes	No
Customer Service	Customer Service	Yes	Yes

Gadgets

Gadget 1 ▲
Customer Lookups

Figure 3: User profile details with Properties query section highlighted

Session Substitution Values

Session Substitution Values are temporary properties that exist while a user is logged into the system. Once the user logs out, the properties are removed. Session Substitution Values can be used anywhere that System Substitution Values can be used. Session Substitution Values are set as a result of running a Data Object Action. See Chapter 6: Data Object Actions for more information about creating Session Substitution Values.

Local vs Global Session Substitution Values

Session Properties are a global bucket that are attached to the user's login session. As such, they are available anywhere and can be set anywhere while the user is logged in. This common bucket gets cleaned out when the user logs out.

For most customers, running an action will typically load almost all its information from the selected record, with Session Properties being used to fill in the gaps. For some ODATA back end systems, you don't really have that option and need to run multiple web service calls at various times against various objects in order to get all the data you need to ultimately do what you need to do. Since these systems do not centralize data where you need it, almost everything is considered a gap that has to be filled with Session Properties. Having multiple tabs open in XHA can be problematic if they all use/set the same Session Property. We have added a new scope attribute to SetSessionProperty to allow the user to have multiple copies of the same Session Property for each window. For example,

```
 <SetSessionProperty>
   <SessionProperty name='S_LSP1' scope='local'><Value>${P_LSP1}</Value></SessionProperty>
</SetSessionProperty>
```

The default scope is "global", which acts the way Session Properties have traditionally. If you instead specify a scope of "local", you will have a new bucket of Local Session Properties for each "window". As you navigate from one window to another, the bucket from window A gets copied to window B. So, each window has a separate bucket of values that is "seeded" from the window they navigated from.

So, imagine you have a Search Action that set a local session property L1 to value A. It then opens a list window which inherits a new copy of L1 to A. It then opens an Action which sets L1 to B. That action has a completion action that opens a detail which inherits a new copy L1 to B. If you then go back past the detail to the original list window, it still has its L1=A value. Each window has its own bucket which is separate from other windows, but when a window is first opened its parent's Local Session Properties are passed over as a base to inherit. **[Note that if an action sets a local session property but has no completion action, the local session property currently gets thrown away. If you want Local Session Properties to get sent back to the parent window, you should have the action do a Close Parent Window and add an Open List/Open Object Completion Action to reopen the original list/details again at the end of the action in order to inherit the new set of Local Session Properties].**

Action Substitution Values

Action Substitution Values exist for a specific Domain Object Action. The following Action Substitution Values are available:

- CURRENT_BARCODE – A Barcode Scanner will be displayed on the device, returning the value scanned

- CURRENT_DATA_OBJECT – The Object Name of the Action's Data Object
- CURRENT_DATA_OBJECT_DOMAIN_CLASS -- The Domain Class of the Action's Data Object
- CURRENT_LOCATION.STREET – A GPS Locator will be displayed on the device, returning the Street Address
- CURRENT_LOCATION.CITY – A GPS Locator will be displayed on the device, returning the City
- CURRENT_LOCATION.STATE – A GPS Locator will be displayed on the device, returning the State
- CURRENT_LOCATION.STATE_CODE – A GPS Locator will be displayed on the device, returning the abbreviated State
- CURRENT_LOCATION.ZIP – A GPS Locator will be displayed on the device, returning the Postal Code
- CURRENT_LOCATION.COUNTY – A GPS Locator will be displayed on the device, returning the County (if available)
- CURRENT_LOCATION.COUNTRY – A GPS Locator will be displayed on the device, returning the Country
- CURRENT_LOCATION.COUNTRY_CODE – A GPS Locator will be displayed on the device, returning the abbreviated Country
- CURRENT_LOCATION.LATITUDE – A GPS Locator will be displayed on the device, returning the normalized Latitude from the device
- CURRENT_LOCATION.LONGITUDE – A GPS Locator will be displayed on the device, returning the normalized Longitude from the device
- CURRENT_LOCATION.LATITUDE_RAW – A GPS Locator will be displayed on the device, returning the actual Latitude from the device
- CURRENT_LOCATION.LONGITUDE_RAW – A GPS Locator will be displayed on the device, returning the actual Longitude from the device
- CURRENT_NFC – An NFC Scanner will be displayed on the device, returning the value scanned
- SELECTED_LOCATION.STREET – The Street Address from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.CITY – The City from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.STATE – The State from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.STATE_CODE – The abbreviated State Address from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.ZIP – The Postal Code from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.COUNTY – The County from the Map's current Drop Pin (if available) will be returned if a Drop Pin exists
- SELECTED_LOCATION.COUNTRY – The Country from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.COUNTRY_CODE – The abbreviated Country from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.LATITUDE – The normalized Latitude from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.LONGITUDE – The normalized Longitude from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_LOCATION.LATITUDE_RAW – The actual Latitude from the Map's current Drop Pin will be returned if a Drop Pin exists

- SELECTED_LOCATION.LONGITUDE_RAW – The actual Longitude from the Map's current Drop Pin will be returned if a Drop Pin exists
- SELECTED_ODATA_OBJECT_ID – The URL path for the selected record (ODATA Data Objects only). See Appendix E: Mapping ODATA $metadata to Xalt for more information about this value.

Note that the above Action Substitution Values (except for SELECTED_ODATA_OBJECT_ID) are only available for messages defined in Data Object Actions (XML Request, Default Value, etc) for mobile clients. For Rich clients running the actions, blanks will be substituted instead.

"Normalized" vs "Raw" latitude and longitude refers to the values returned in the respective substitution values above. The "Raw" values are the actual lat/long captured by the device itself. This lat/long is then sent to an address geolocator in order to retrieve the address components (Street, City, State, etc) for this location. As part of this process, the lat/long are normalized to point to the center of the address. The "normalized" substitution values return this lat/long.

Data Source Substitution Values

Data Source Substitution Values exist for a specific Data Source. The following Data Source Substitution Values are available:

- CURRENT_DATA_SOURCE – The Description for the Data Source
- CURRENT_ENVIRONMENT – The Environment ID or Primary Library Name for the Data Source
- CURRENT_PRIMARY_USER – The Primary User ID for the Data Source
- CURRENT_SYSTEM – The System Name for the Data Source
- CURRENT_SYSTEM_URL – The System URL for the Data Source

CURRENT_SYSTEM_URL is typically used when performing an ExecuteHTTPAction in an Action Request XML. This allows you to soft-code the Base URL in cases where your Action is rooted at the same URL as the System URL (so that later changes to a new System URL will not require you to change all your actions). We also allow a special function ${CURRENT_SYSTEM_URL-x} to be able to call URLs that are rooted in parent folders of the System URL. For example, if the Data Source System URL is:

https://ppm.intergraph.com/SPFNPAPServer/API/v2/Hexagon

${CURRENT_SYSTEM_URL} = https://ppm.intergraph.com/SPFNPAPServer/API/v2/Hexagon
${CURRENT_SYSTEM_URL-1} = https://ppm.intergraph.com/SPFNPAPServer/API/v2
${CURRENT_SYSTEM_URL-2} = https://ppm.intergraph.com/SPFNPAPServer/API
${CURRENT_SYSTEM_URL-3} = https://ppm.intergraph.com/SPFNPAPServer

and so on.

Note that Data Source Substitution Values are only available for messages defined in Data Object Actions (XML Request, Default Value, etc.).

Data Object Substitution Values

Every property for a Data Object can be used as a substitution value. For example, Data Object's descriptive title for Orders can be: **Order: ${orderNumber}**. At runtime, the orderNumber field for the selected Order will be substituted, and the end user will see: **Order: PO0322**. Note that substitution values are case-sensitive, so be sure to enter the property name EXACTLY as it appears in the Defined Properties section of the Data Object.

Appendix B: Specifying Where Clauses

There are a number of places where Xalt allows you to specify where clauses to help filter the data that is returned. In general, these where clauses are passed directly to the back-end system, so you should specify these values using a syntax acceptable to the back-end system. Some common examples are

((order_status > '30' and order_status <= '50') or order_status = '99')

(order_amount > 10000)

Note that you should not include the "where" constant for these clauses, as it will be added automatically by Xalt when formulating the query.

You can also use substitution values when formulating the query. Standard, Default, and User Substitution Values are available at all times. Data Object Substitution Values are only available in cases where a Data Object is used.

There are a few places, such as Detail Query Section and Data Object Lookup where multiple Data Objects come into play. For example, in an SQL Data Source when displaying a list of related Order Items in a Detail Query Section for Order, if you leave the Additional Where Clause blank, the list will show you ALL Order Items for ALL Orders, which is probably not what you intended. You only want to see the Order Items for the order that you are viewing. In this case, you would have the following Additional Where Clause:

'${FROM_OBJECT.orderNumber}' = orderNumber

In General, when you specify a property name, Xalt assumes the property is for the current Data Object being queried. In this case, we have 2 Data Objects: The Data Object we are looking at (Order) and the Data Object we are querying the list for (Order Item). For substitution values in these situations, prefix the name with FROM_OBJECT. to specify that you want the orderNumber property from the Order Data Object. When Xalt formulates the final query, the above clause will be transformed to something like the following:

'11002' = orderNumber

One final note to consider comes from field name ambiguity. Again, let's take SQL Data Sources as an example. Notice that both the Order and the Order Item tables contain a field name orderNumber. If you specify orderNumber in an SQL statement in which you are querying both Order and Order Item database tables, SQL will throw an error stating that "orderNumber" is ambiguous because it does not know which table to get the field from. Xalt resolves this ambiguity in most cases by appending the table name to the field. In the cases of where clauses, Xalt does only a minimal amount of alteration of the clause before passing it to the back-end system. Let's say you have an Additional Where Clause as follows for a Detail Query Section off Order:

'${FROM_OBJECT.orderNumber}' = orderNumber AND relatedOrder.status <> '99'

When Xalt converts this, the SQL will look something like:

SELECT …. FROM order_item order_item

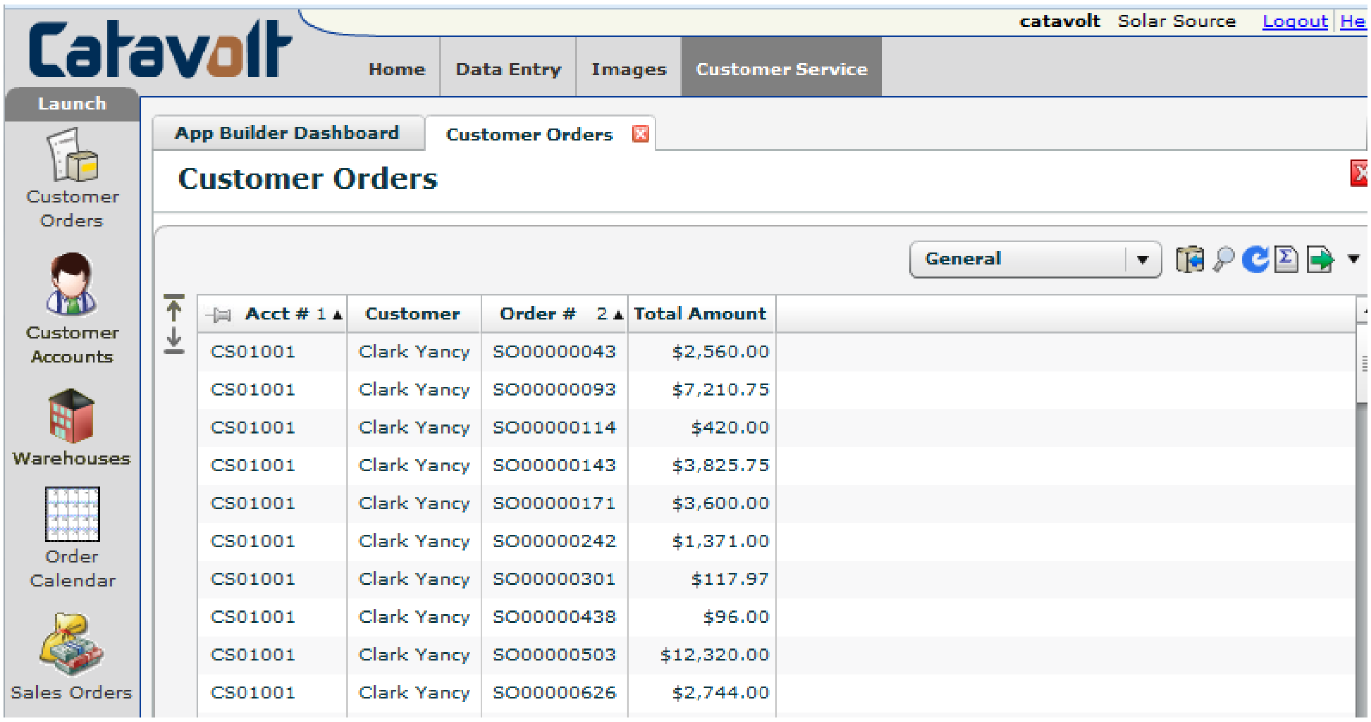
LEFT OUTER JOIN order_header relatedOrder on order_item.orderNumber = relatedOrder.orderNumber

WHERE '11002' = orderNumber and relatedOrder.status <> '99'

It is likely that you will get the ambiguous error for orderNumber because you did not specify which table it is from. Changing **orderNumber** to **order_item.orderNumber** (prepend <tableName>. to the field name) will resolve the ambiguity. Note that pre-pending the table name to fields in where clauses is only required in cases where you are querying multiple tables that have common field names.

Appendix C: Summary (Group-By) Queries

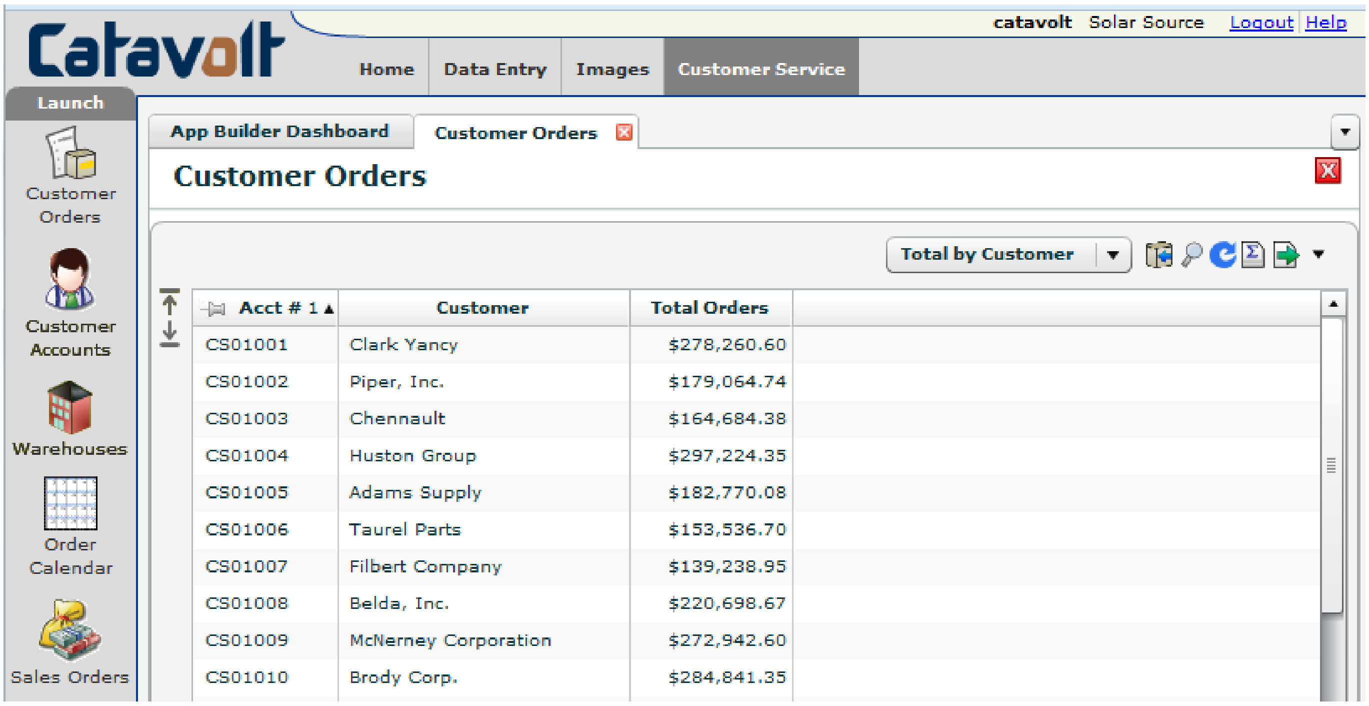
There are times when you may wish to be able to summarize data in your database tables. For example, you may have a Customer Order table that contains the following fields:



Customer Orders

Acct # 1 ▲	Customer	Order # 2 ▲	Total Amount
CS01001	Clark Yancy	SO00000043	$2,560.00
CS01001	Clark Yancy	SO00000093	$7,210.75
CS01001	Clark Yancy	SO00000114	$420.00
CS01001	Clark Yancy	SO00000143	$3,825.75
CS01001	Clark Yancy	SO00000171	$3,600.00
CS01001	Clark Yancy	SO00000242	$1,371.00
CS01001	Clark Yancy	SO00000301	$117.97
CS01001	Clark Yancy	SO00000438	$96.00
CS01001	Clark Yancy	SO00000503	$12,320.00
CS01001	Clark Yancy	SO00000626	$2,744.00

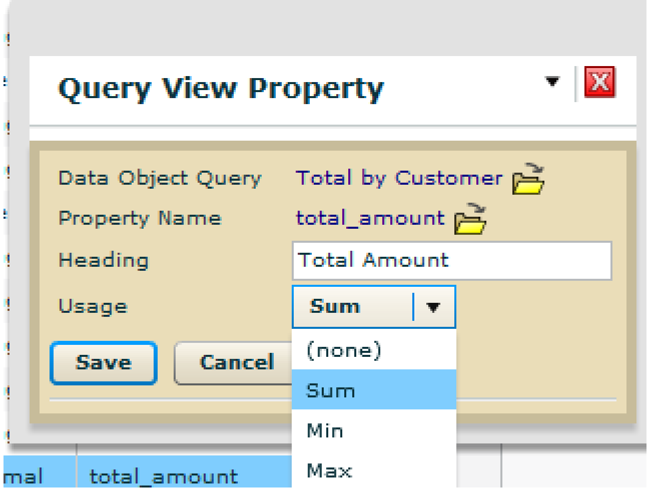
You would like to be able to display the total amount of all orders per customer:



Customer Orders

Acct # 1 ▲	Customer	Total Orders
CS01001	Clark Yancy	$278,260.60
CS01002	Piper, Inc.	$179,064.74
CS01003	Chennault	$164,684.38
CS01004	Huston Group	$297,224.35
CS01005	Adams Supply	$182,770.08
CS01006	Taurel Parts	$153,536.70
CS01007	Filbert Company	$139,238.95
CS01008	Belda, Inc.	$220,698.67
CS01009	McNerney Corporation	$272,942.60
CS01010	Brody Corp.	$284,841.35

In previous releases, you would have to create a Data Object over an SQL Summary View in order to achieve this. Now, you simply add a Usage of Sum when adding the Total Amount property to the Query:



Query View Property

Data Object Query: Total by Customer

Property Name: total_amount

Heading: Total Amount

Usage: Sum

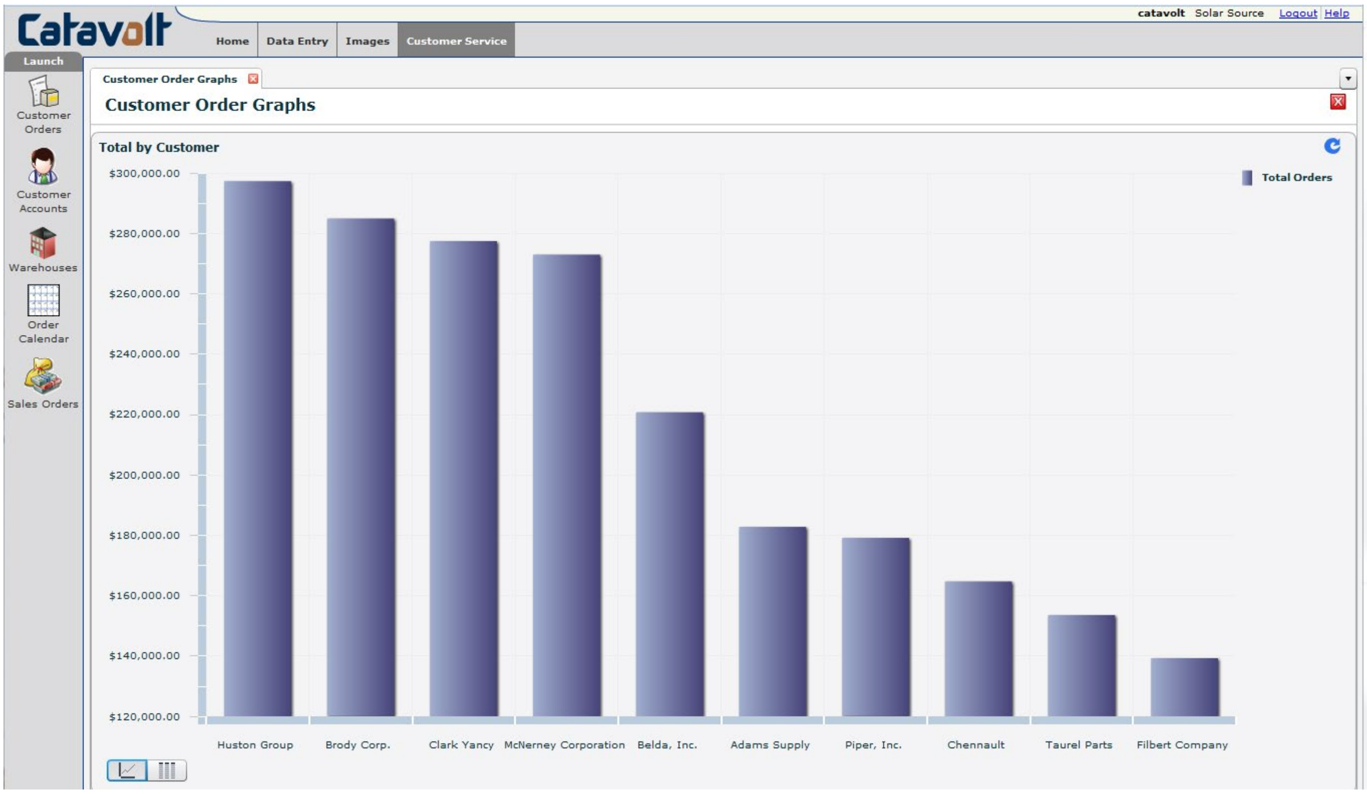
For those familiar with SQL, we are utilizing Aggregate Functions in conjunction with Group By and Having clauses in order to fulfill the requested function. While it may seem like a seamless operation to change a normal Query into a Summary Query, there are a number of things to consider.

- Sum, Min, and Max are the currently available aggregate Usage values. They are available for numeric properties.
- As this is an SQL function, these Usage values are only available for Data Objects in Direct SQL and IBM i Data Sources.
- Any field you add to the Query that does not have an aggregate Usage value will be automatically considered a Group By field.
- You can only Sort on properties that you have added to the Query (i.e. in the View Properties list). Any other fields that you have chosen will be ignored. This is because SQL restricts the Order By clause to only have fields with aggregate functions or group-by fields.
- Currently, you cannot add the same property to the View Properties list more than once. This means that you cannot add the same property with multiple aggregate usages (say, Sum and Max) to the same Query. You can work around this by creating a logical property whose calculation is simply the property value. This will give you a logical duplicate property that you can add to a Query and have a different aggregate usage.
- In most cases (Sort fields, Data Annotation Rules, etc.), the values you will be manipulating will be the aggregate values. The exception is in Where clauses (Query Where clause, Query Section Additional Where, etc.). Again, in SQL, the Where Clause is performed before the Group By clause, meaning that the records are selected out before they are grouped. Because of this, a Where Clause of “total_amount > 3000” will only group records where total_amount > 3000. It will NOT exclude any grouped records where the Sum aggregate value is > 3000.
- You cannot run the Open/Change/Delete/Select actions on a selected Summary record. Because Summary records are by definition a summary of multiple records and not individual database table records, object-level functions such as Change and Delete are not appropriate. We will display an error message if you attempt to run one of these actions against a Summary record.

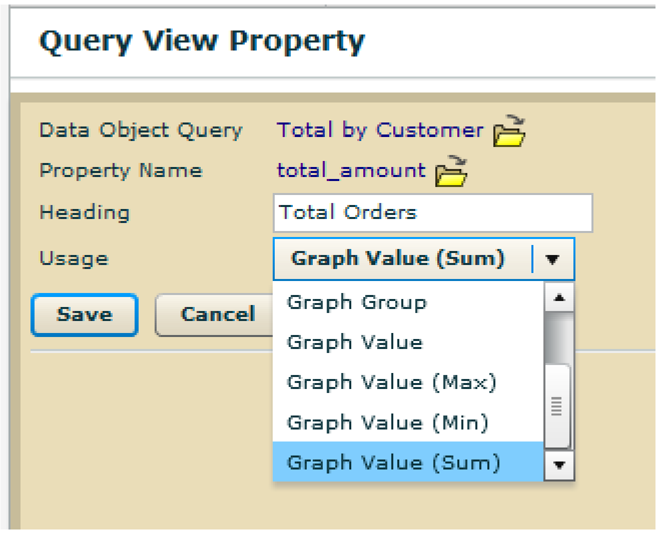
- If you run an Action against a Summary record and the action refers to properties from the selected record, only values for Group by fields (fields in the View Properties list that do not have aggregate usages) are available. This is again due to the fact that Summary records are not individual database table records.
- Normally on an Object Lookup query for the combo box values, the referenced Query is used for its Where Clause and Sort value, but the list of columns to use is ignored (since the Lookup has its own list of values to query). If you use a Summary Query in an Object Lookup, the query columns will be included. Note that in addition to these fields, any fields referenced in the Lookup (Lookup Property, fields referenced in Combo Box Display Format, and Additional Side Effect fields) will also be added to the Query and will be included in the Group By clause.
- If you use a Summary Query in an Object Lookup, the Find (magnifying glass) button is not available, as the Select action on the Find list is not allowed.

You can mix regular and Summary Queries in the same Data Object. In addition, you can also implement drill-down behavior by using no-prompt actions with Open List redirections as a Query's Default Action. With this technique, you can drill from one Summary Query into a more detailed Summary Query, you can drill from a Summary Query into a regular Query, etc.

You can also create a Graph over a Summary Query, for example:



As in Summary Queries for tables, you simply add a Usage of Graph Value (Sum) when adding the Total Amount property to the Query:



Query View Property

Data Object Query: Total by Customer
Property Name: total_amount
Heading: Total Orders
Usage: Graph Value (Sum)

The same considerations above for Table Summary Queries apply for Graphs, in addition to the following:

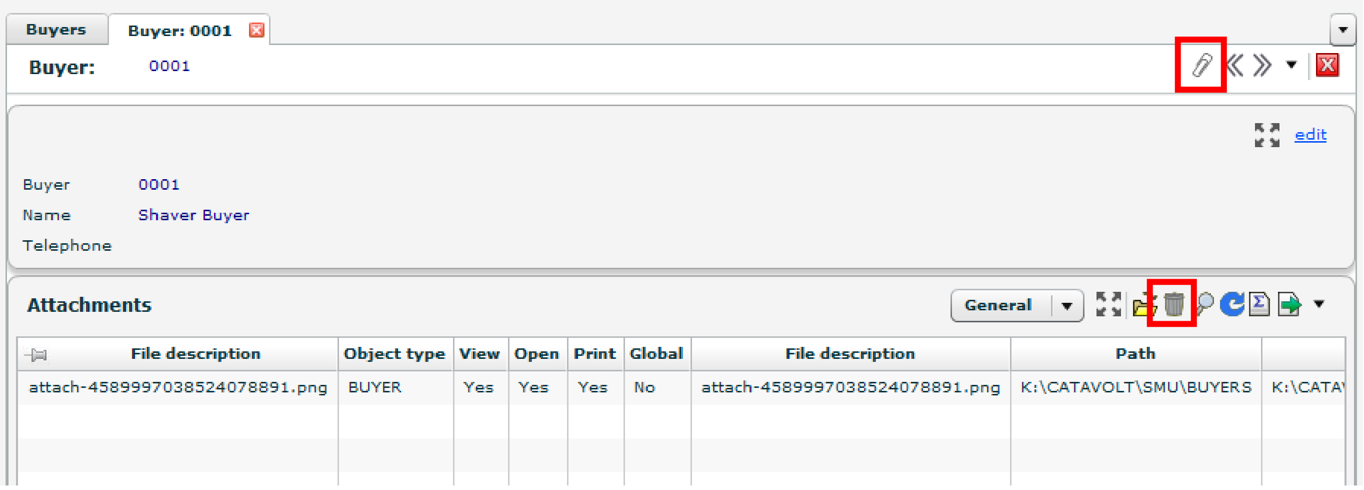
- Graph Value (Sum), Graph Value (Min), and Graph Value (Max) are the currently available aggregate Usage values. They are available for numeric properties.
- You cannot mix the above values with Graph Value. Either all value properties should be marked as Graph Value or be marked with one of the Summary values above.
- You cannot mix the above values with Graph Group. If you are using Graph Group, you cannot use the Summary values above, as Summary value have grouping built-in.
- Currently, if you do not mark any properties as Identifier properties, the Graph will use the Short Title field in the Data Object to identify the record. You can only use properties in the Short Title that you have added to the Query (i.e. in the View Properties list). Any other fields that you have chosen will be replaced with blanks. This is because SQL restricts the select clause to only have fields with aggregate functions or group-by fields.

Appendix D: Attachments

There are times when you may wish to attach data files to records in your Data Object. For example, you may want to attach a picture of an Item to its corresponding Item record, or attach a PDF Invoice to a Customer Order.

Infor XA Attachments

Infor XA supports the concept of attaching files to data records within System-Link. Xalt uses System-Link to access attachment files as well as being able to create and/or delete attachments. Creating an Attachments Query Section will automatically create a paperclip button to allow you to attach a file to the record. You will also get a trashcan button to allow you to Delete an attachment on the attachments list if it is configured. Double-clicking on an attachment record will download and open the data file associated with the attachment.



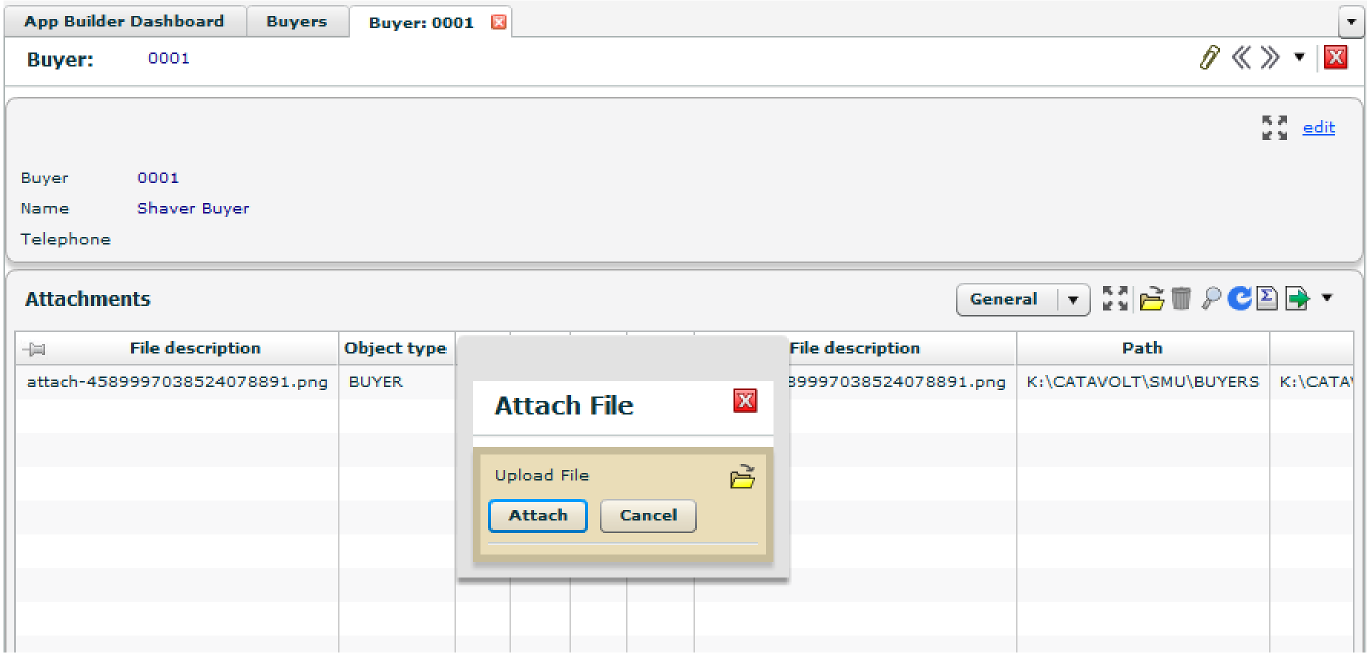
Buyer: 0001

Buyer 0001
Name Shaver Buyer
Telephone

Attachments

File description	Object type	View	Open	Print	Global	File description	Path	
attach-4589997038524078891.png	BUYER	Yes	Yes	Yes	No	attach-4589997038524078891.png	K:\CATAVOLT\SMU\BUYERS	K:\CATA

Clicking the create action will allow you to upload a file as a new Attachment from a rich or mobile client:



Buyer: 0001

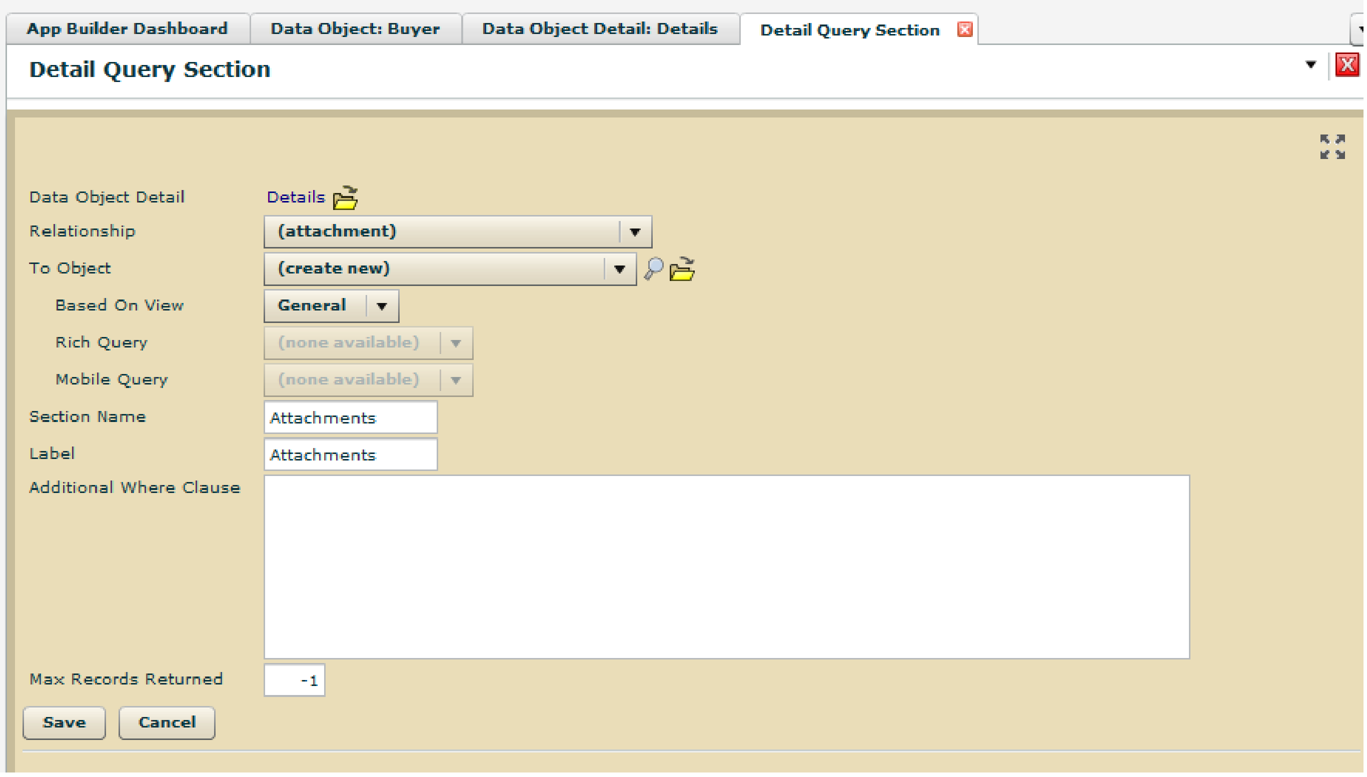
Buyer 0001
Name Shaver Buyer
Telephone

Attachments

File description	Object type					File description	Path	
attach-4589997038524078891.png	BUYER					89997038524078891.png	K:\CATAVOLT\SMU\BUYERS	K:\CATA

Attach File

To create an Attachments section, first make sure that Allow Attachments is set to Yes for your Data Object. Create a new Query Section:

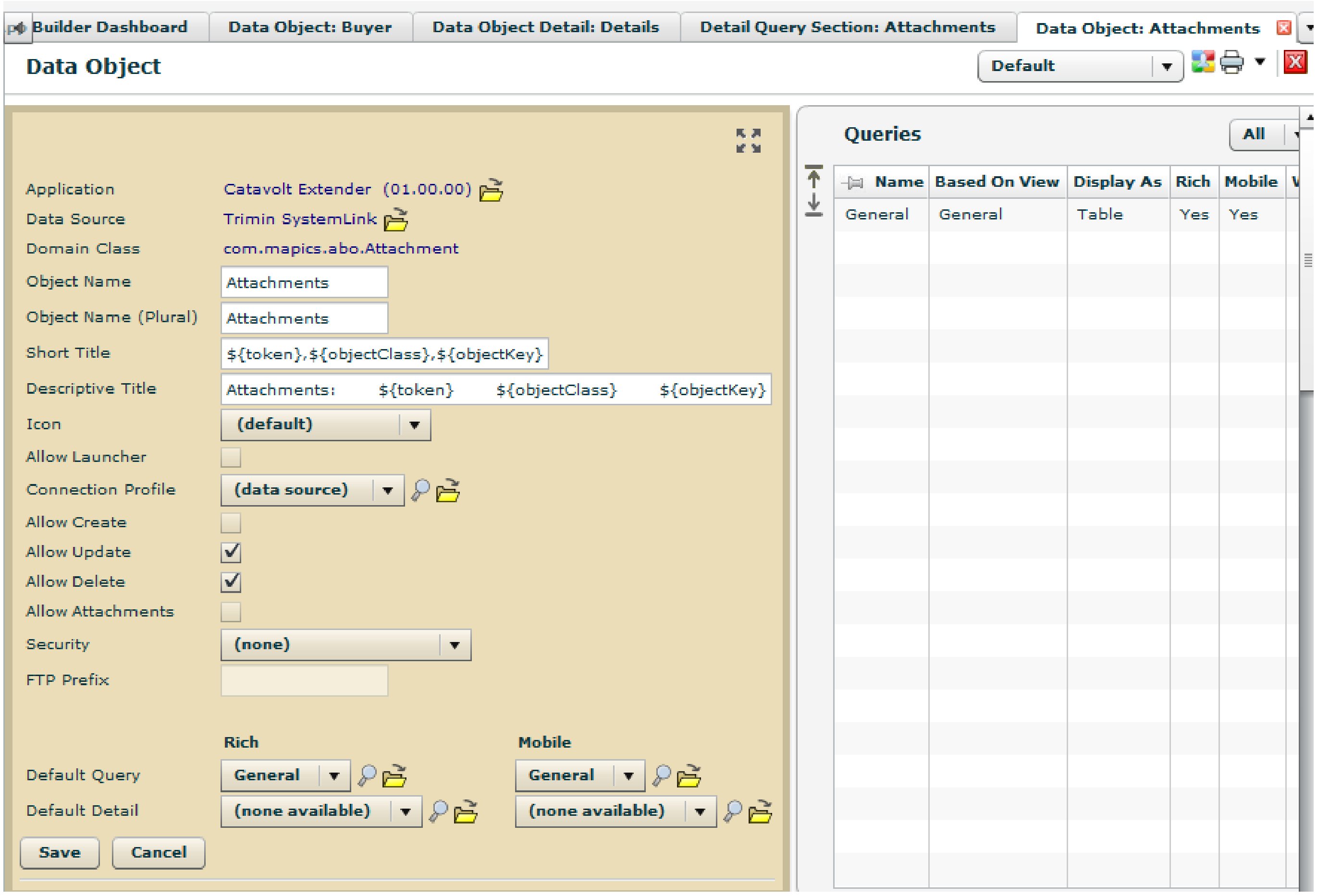


Detail Query Section

Data Object Detail	Details
Relationship	(attachment)
To Object	(create new)
Based On View	General
Rich Query	(none available)
Mobile Query	(none available)
Section Name	Attachments
Label	Attachments
Additional Where Clause	
Max Records Returned	-1

The Relationship list will contain a special value called (attachment) that will allow you to show a list of Attachments for the related record. To To Object will allow you to choose from an existing Attachment Data Object or allow you to create a new one. The rest of the section is filled out as normal.

Creating an Attachment section will automatically give you a create action to add new files as Attachments. To allow deletes, you can select the Allow Delete open on the Attachment object. Leaving this box unchecked will remove the Delete action from the menu.



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	Trimin SystemLink
Domain Class	com.mapics.abo.Attachment
Object Name	Attachments
Object Name (Plural)	Attachments
Short Title	${token},${objectClass},${objectKey}
Descriptive Title	Attachments: ${token} ${objectClass} ${objectKey}
Icon	(default)
Allow Launcher	□
Connection Profile	(data source)
Allow Create	□
Allow Update	☑
Allow Delete	☑
Allow Attachments	□
Security	(none)
FTP Prefix	

	Rich	Mobile
Default Query	General	General
Default Detail	(none available)	(none available)

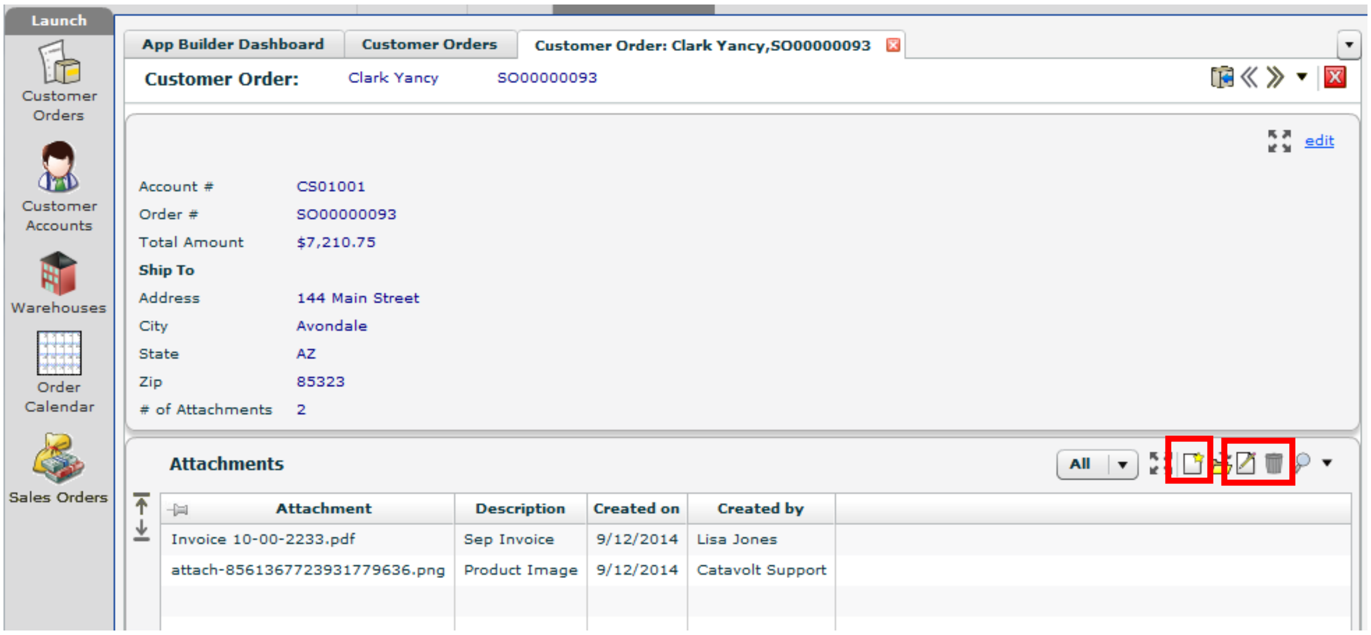
Queries

Name	Based On View	Display As	Rich	Mobile
General	General	Table	Yes	Yes

Attachments for Other Data Sources

For other Data Source types, Xalt manages associating data files with data records. Xalt allows you to access attachment files as well as being able to create, update, and/or delete attachments.

To view an existing Attachment, double-click an attachment record or select it and choose the Open menu action. Xalt will download and open the data file associated with the attachment. You may also download multiple Attachments at the same time by selecting multiple records and pressing the Open menu action. Xalt will gather the individual data files into a .zip file and then download the .zip file to your browser.



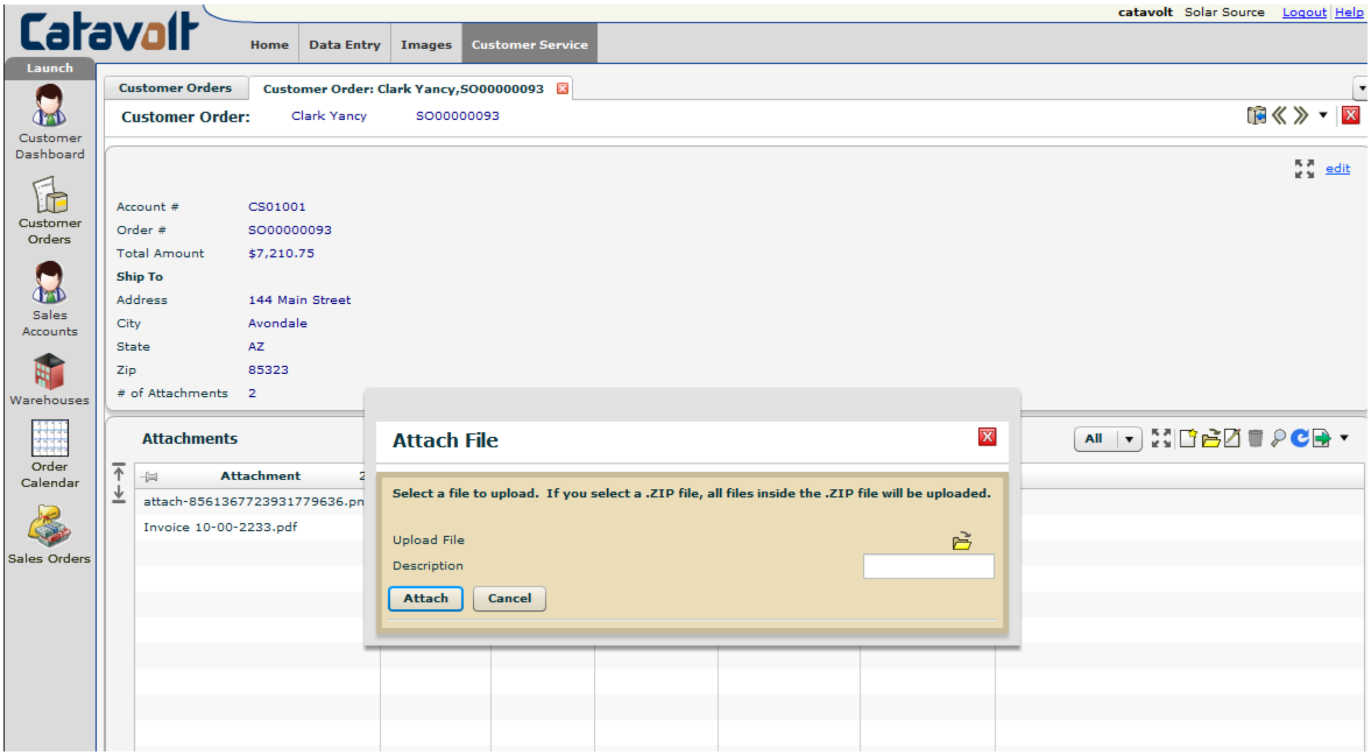
Customer Order: Clark Yancy SO00000093

Account # CS01001
Order # SO00000093
Total Amount $7,210.75
Ship To
Address 144 Main Street
City Avondale
State AZ
Zip 85323
# of Attachments 2

Attachments

Attachment	Description	Created on	Created by
Invoice 10-00-2233.pdf	Sep Invoice	9/12/2014	Lisa Jones
attach-8561367723931779636.png	Product Image	9/12/2014	Catavolt Support

Clicking the create action will allow you to upload a file as a new Attachment from a rich or mobile client:



Customer Order: Clark Yancy SO00000093

Account #	CS01001
Order #	SO00000093
Total Amount	$7,210.75
Ship To	
Address	144 Main Street
City	Avondale
State	AZ
Zip	85323
# of Attachments	2

Attachment
attach-8561367723931779636.pn
Invoice 10-00-2233.pdf

Attach File

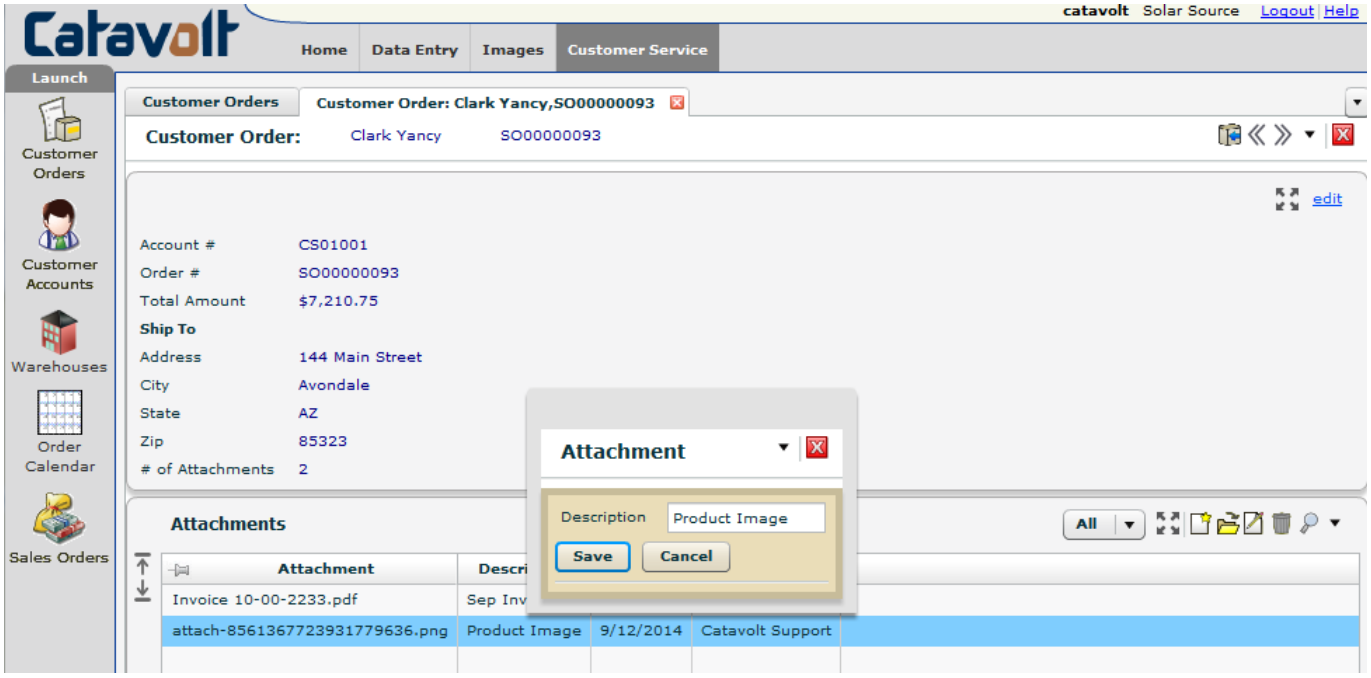
Select a file to upload. If you select a .ZIP file, all files inside the .ZIP file will be uploaded.

Upload File

Description

Specifying a regular file will create an Attachment record for that file. You can also attach multiple files at one time by specifying a .zip file. If you specify a .zip file, Xalt will unzip the contents and do an individual Attachment create for each file inside the .zip file (note that any path information inside the .zip file will be ignored during the Attachment create).

Updating an Attachment allows you to change the Attachment's description:



Customer Order: Clark Yancy SO00000093

Account #	CS01001
Order #	SO00000093
Total Amount	$7,210.75
Ship To	
Address	144 Main Street
City	Avondale
State	AZ
Zip	85323
# of Attachments	2

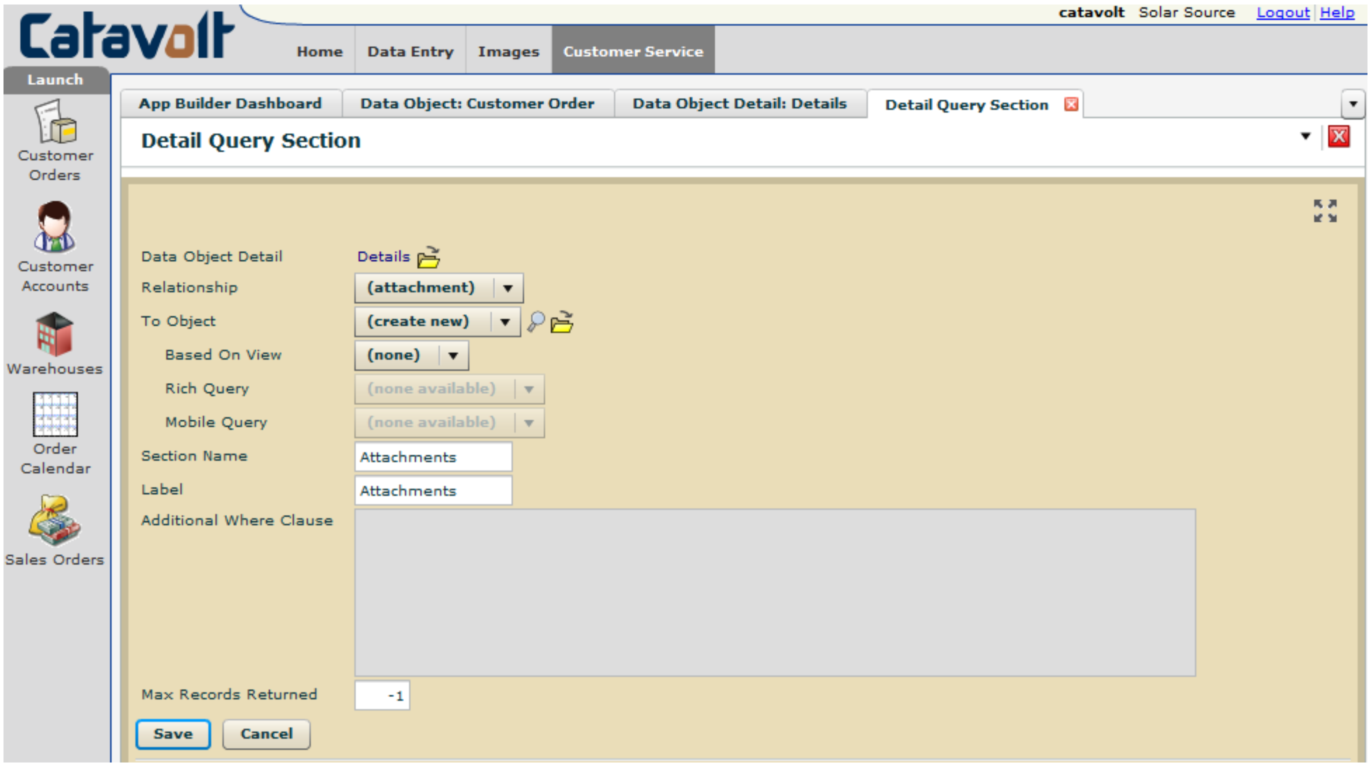
Attachment	Descri		
Invoice 10-00-2233.pdf	Sep Inv		
attach-8561367723931779636.png	Product Image	9/12/2014	Catavolt Support

Attachment

Description: Product Image

Deleting an attachment will remove the data file and the Attachment record.

To create an Attachments section, first make sure that Allow Attachments is set to Yes for your Data Object. Create a new Query Section:

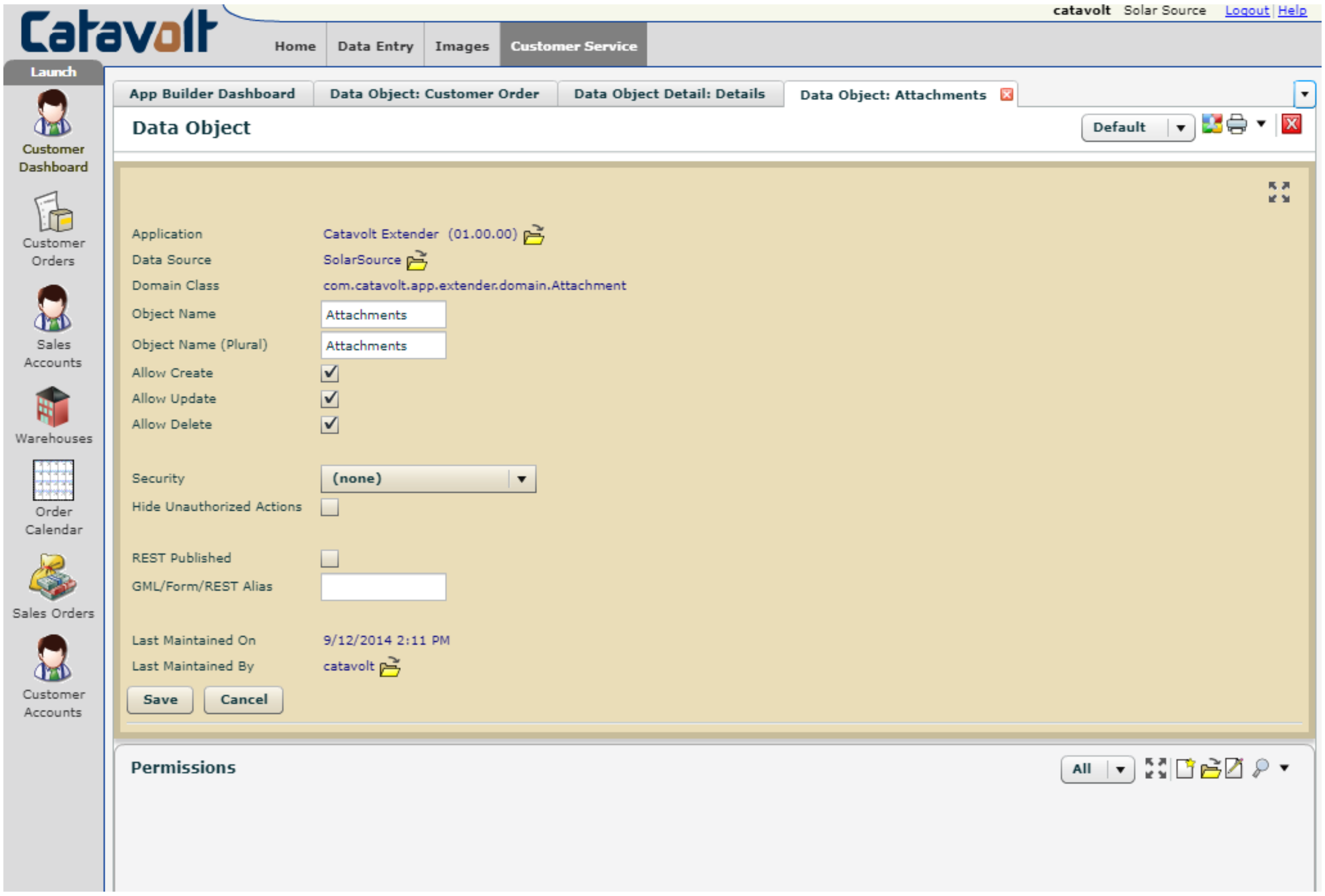


Detail Query Section

Data Object Detail	Details
Relationship	(attachment)
To Object	(create new)
Based On View	(none)
Rich Query	(none available)
Mobile Query	(none available)
Section Name	Attachments
Label	Attachments
Additional Where Clause	
Max Records Returned	-1

The Relationship list will contain a special value called (attachment) that will allow you to show a list of Attachments for the related record. To To Object will allow you to choose from an existing Attachment Data Object or allow you to create a new one. The rest of the section is filled out as normal.

Looking at the Attachment Data Object that is created gives you a stripped-down data object. You can control whether the Create, Update, and/or Delete actions appear using the Allow Create, Allow Update, and Allow Delete options. You use the Security option to set up Role-based security for Attachments. You can also expose the Attachments object to the Hexagon REST API using the REST Published and GML/Form/REST Alias options (See the Xalt | Mobility REST API document for more information).



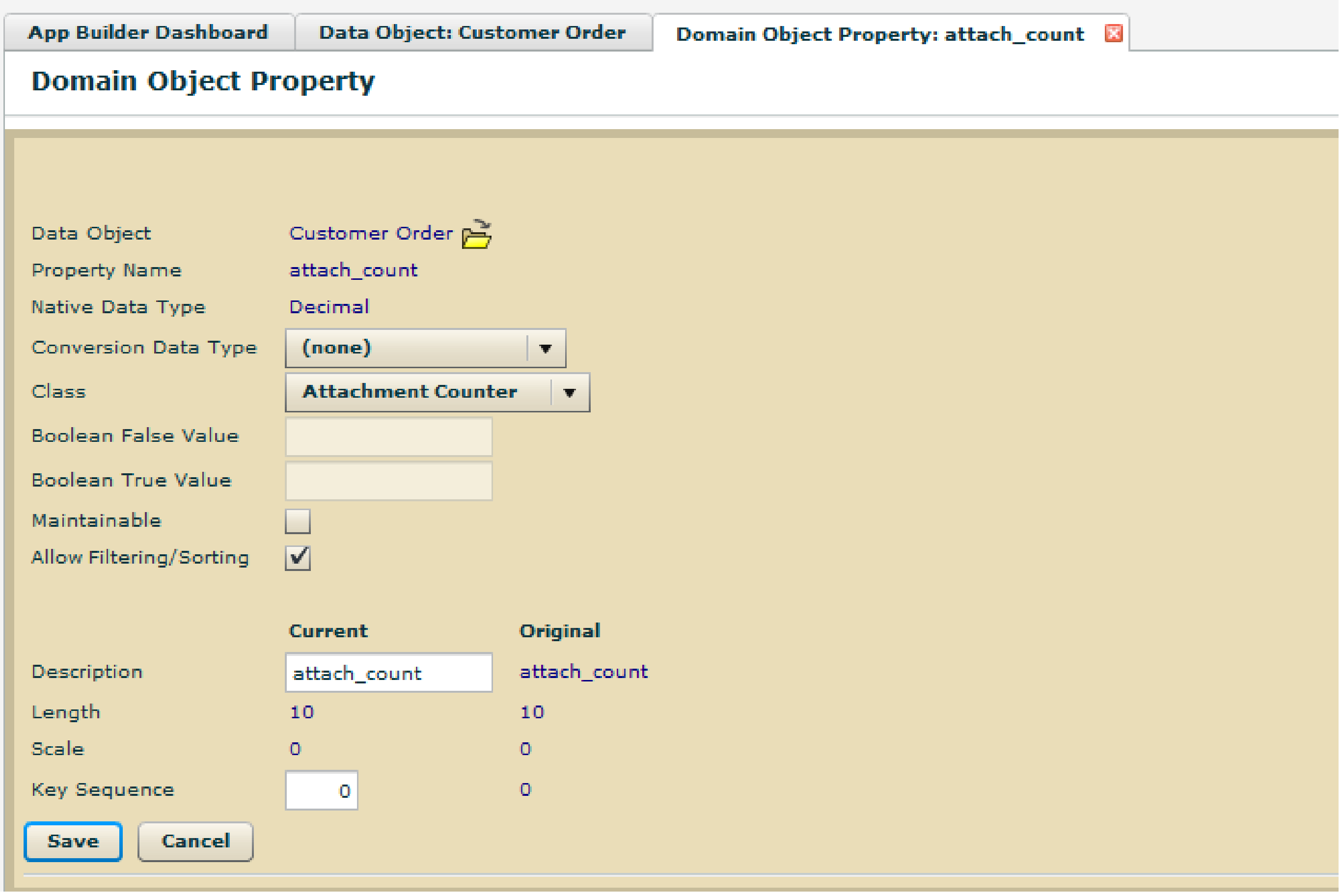
Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SolarSource
Domain Class	com.catavolt.app.extender.domain.Attachment
Object Name	Attachments
Object Name (Plural)	Attachments
Allow Create	☑
Allow Update	☑
Allow Delete	☑
Security	(none)
Hide Unauthorized Actions	☐
REST Published	☐
GML/Form/REST Alias	
Last Maintained On	9/12/2014 2:11 PM
Last Maintained By	catavolt

Save Cancel

Permissions

You can also designate a property on the owning Data Object as an Attachment Counter. As Attachments are created and deleted by Xalt, it will update this property on your Data Object to identify the number of Attachments currently connected to the record. To do this, set the Class value on the property to Attachment Counter. Note that the property type must be numeric in order for this option to appear.


App Builder Dashboard
Data Object: Customer Order
Domain Object Property: attach_count

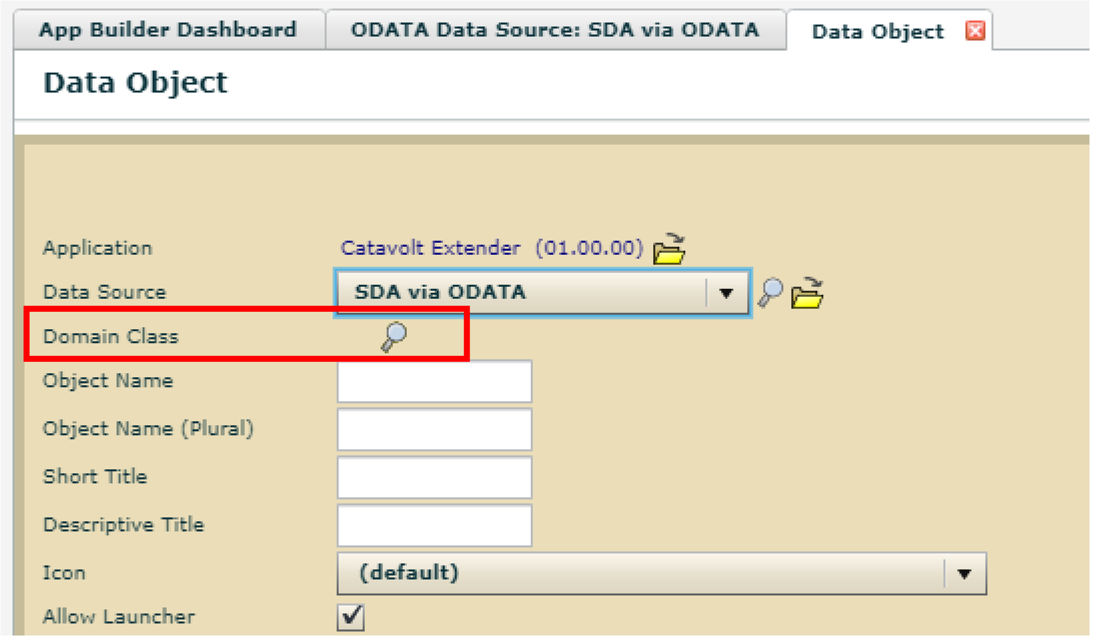
Domain Object Property

Data Object	Customer Order	
Property Name	attach_count	
Native Data Type	Decimal	
Conversion Data Type	(none)	
Class	Attachment Counter	
Boolean False Value		
Boolean True Value		
Maintainable	□	
Allow Filtering/Sorting	☑	
	Current	**Original**
Description	attach_count	attach_count
Length	10	10
Scale	0	0
Key Sequence	0	0

Appendix E: Mapping ODATA $metadata to Xalt

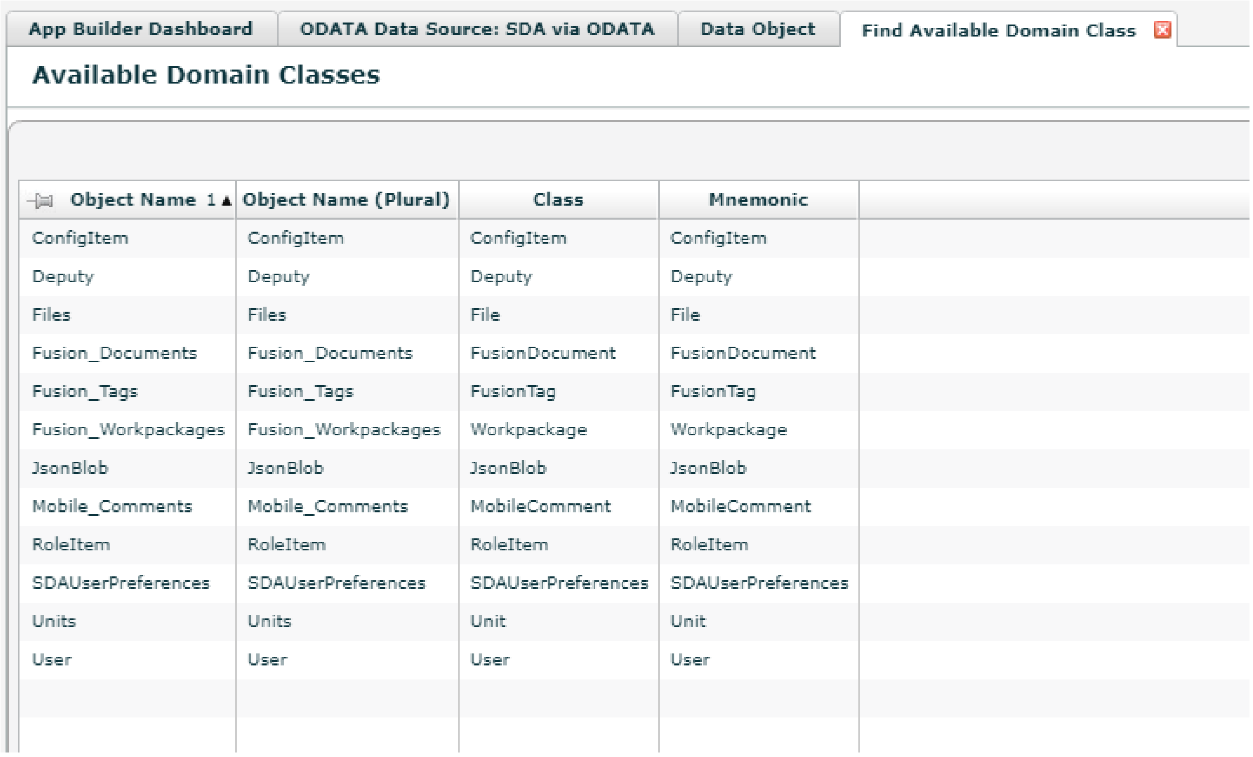
Data Objects

When creating a new Data Object, Xalt shows you a list of Available Domain Classes



Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SDA via ODATA
Domain Class	
Object Name	
Object Name (Plural)	
Short Title	
Descriptive Title	
Icon	(default)
Allow Launcher	☑



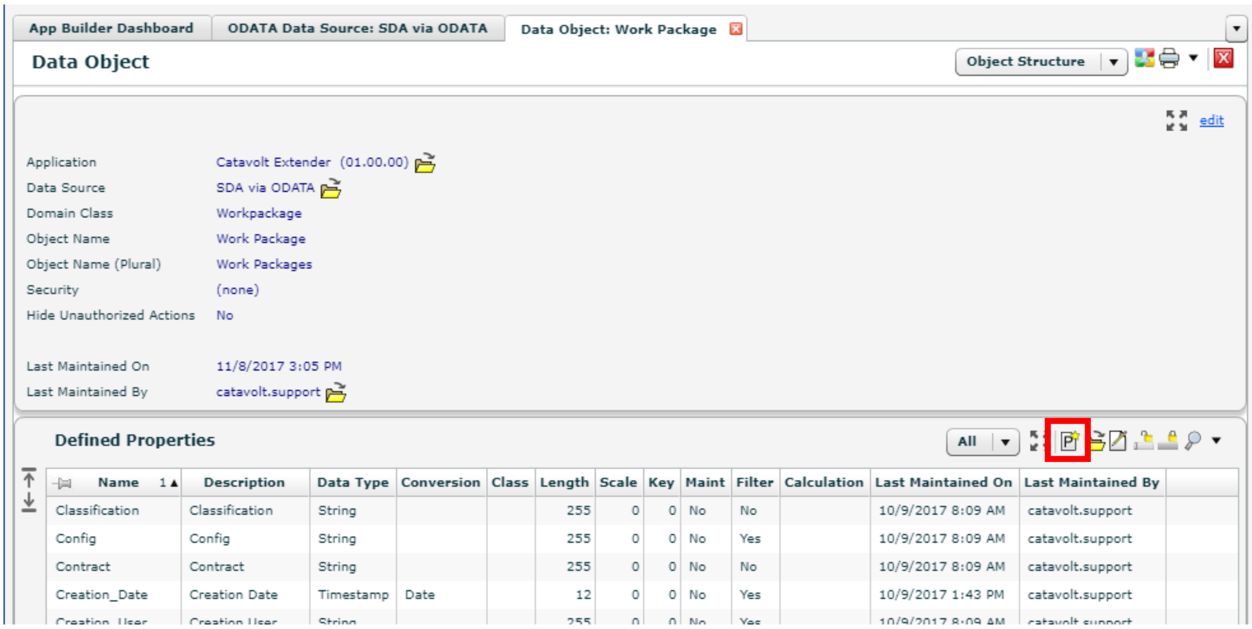
Available Domain Classes

Object Name 1 ▲	Object Name (Plural)	Class	Mnemonic
ConfigItem	ConfigItem	ConfigItem	ConfigItem
Deputy	Deputy	Deputy	Deputy
Files	Files	File	File
Fusion_Documents	Fusion_Documents	FusionDocument	FusionDocument
Fusion_Tags	Fusion_Tags	FusionTag	FusionTag
Fusion_Workpackages	Fusion_Workpackages	Workpackage	Workpackage
JsonBlob	JsonBlob	JsonBlob	JsonBlob
Mobile_Comments	Mobile_Comments	MobileComment	MobileComment
RoleItem	RoleItem	RoleItem	RoleItem
SDAUserPreferences	SDAUserPreferences	SDAUserPreferences	SDAUserPreferences
Units	Units	Unit	Unit
User	User	User	User

Xalt extracts all <EntityType> tags from $metadata in order to build this list.

Properties

When defining a Data Object, Xalt shows you the list of Available Properties

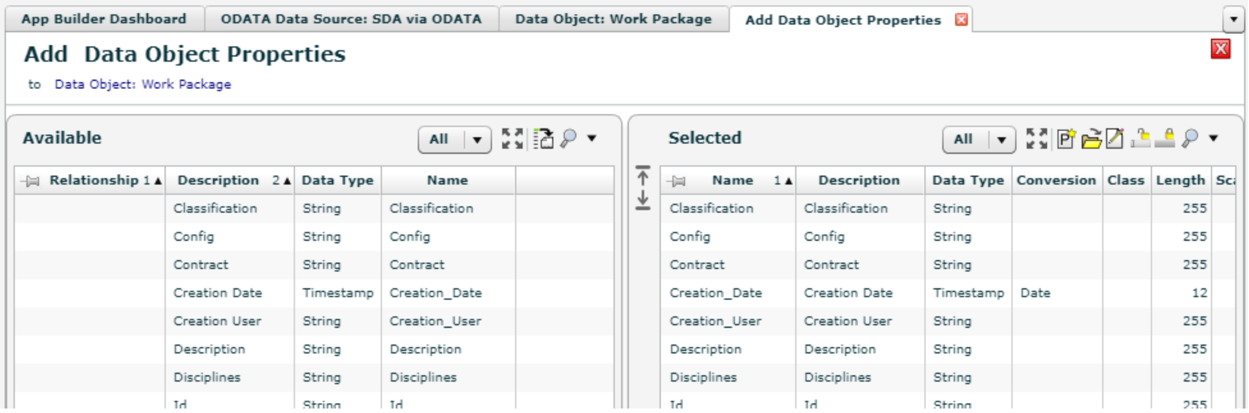


Data Object

Application	Catavolt Extender (01.00.00)
Data Source	SDA via ODATA
Domain Class	Workpackage
Object Name	Work Package
Object Name (Plural)	Work Packages
Security	(none)
Hide Unauthorized Actions	No
Last Maintained On	11/8/2017 3:05 PM
Last Maintained By	catavolt.support

Defined Properties All

Name 1▲	Description	Data Type	Conversion	Class	Length	Scale	Key	Maint	Filter	Calculation	Last Maintained On	Last Maintained By
Classification	Classification	String			255	0	0	No	No		10/9/2017 8:09 AM	catavolt.support
Config	Config	String			255	0	0	No	Yes		10/9/2017 8:09 AM	catavolt.support
Contract	Contract	String			255	0	0	No	No		10/9/2017 8:09 AM	catavolt.support
Creation_Date	Creation Date	Timestamp	Date		12	0	0	No	Yes		10/9/2017 1:43 PM	catavolt.support
Creation_User	Creation User	String			255	0	0	No	Yes		10/9/2017 8:09 AM	catavolt.support



App Builder Dashboard
ODATA Data Source: SDA via ODATA
Data Object: Work Package
Add Data Object Properties

Add Data Object Properties

to Data Object: Work Package

Available All

Relationship 1▲	Description 2▲	Data Type	Name
	Classification	String	Classification
	Config	String	Config
	Contract	String	Contract
	Creation Date	Timestamp	Creation_Date
	Creation User	String	Creation_User
	Description	String	Description
	Disciplines	String	Disciplines
	Id	String	Id

Selected All

Name 1▲	Description	Data Type	Conversion	Class	Length	Sc
Classification	Classification	String			255	
Config	Config	String			255	
Contract	Contract	String			255	
Creation_Date	Creation Date	Timestamp	Date		12	
Creation_User	Creation User	String			255	
Description	Description	String			255	
Disciplines	Disciplines	String			255	
Id	Id	String			255	

Xalt extracts the <Property> tags inside an EntityType to build the list of properties. Complex Property Types are recursively flattened out into individual properties (For example, an Address complex property will turn into multiple simple properties in Xalt such as Address/Street, Address/City, etc). In the case of Inherited types, Xalt will also recursively pull properties from base types.

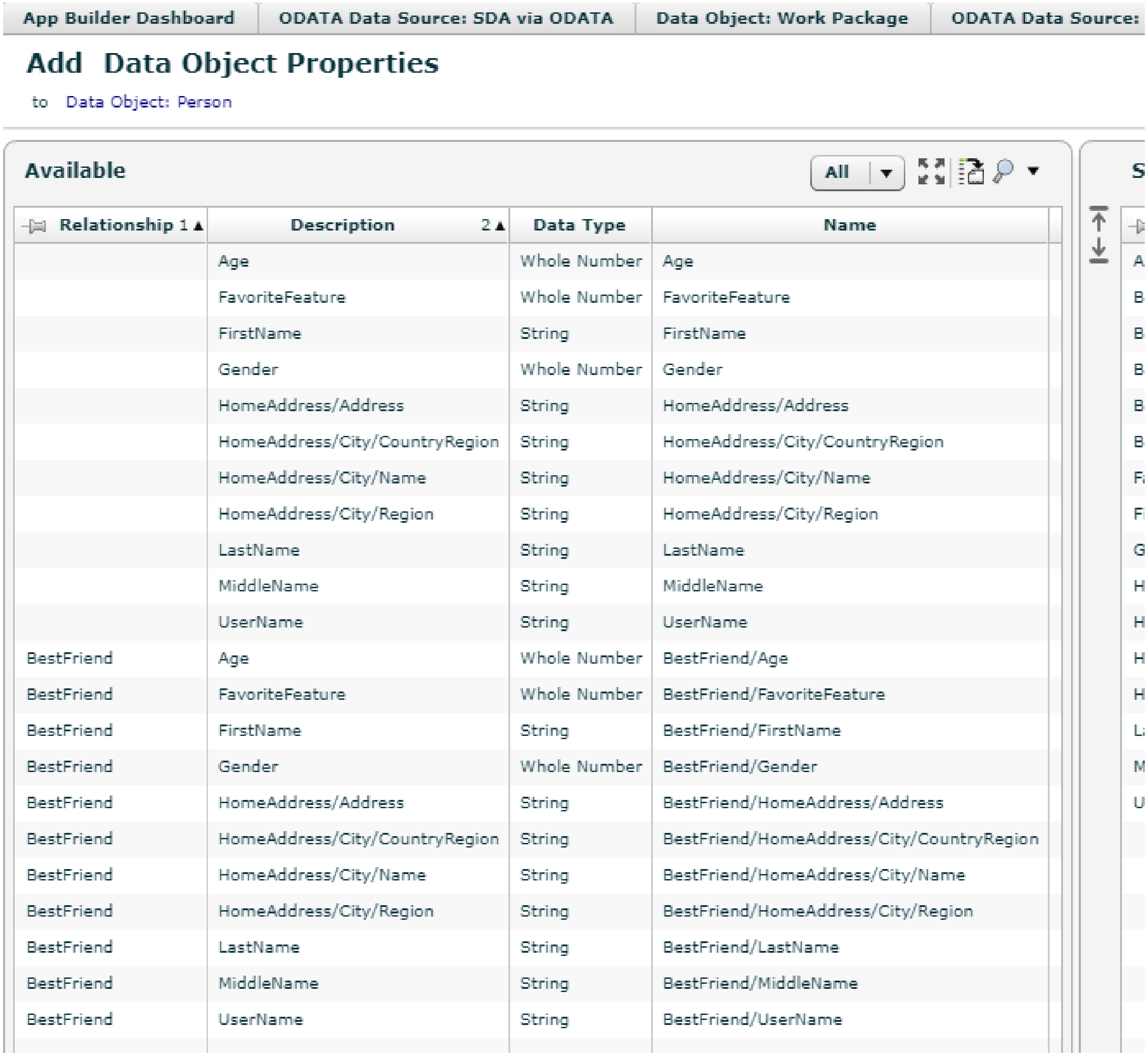
Logical Properties

ODATA supports a concept called OpenType, which means an EntityType that contains dynamic properties. Dynamic properties are properties that can be read and/or written to but do not exist in the EntityType specification. Xalt adds supports for OpenType by allowing you to create Logical Properties for ODATA Data objects. These differ from normal Logical Properties as follows:

- The Calculation property is disabled. These values will be returned directly by the ODATA call and as such a Calculation is not needed.
- The Maintainable property is enabled. These values have the option to be sent (written) as well as received (read), based on the specific ODATA object.

Related Data Objects

Xalt also extracts <NavigationProperty> tags inside an EntityType. If the Navigation is to a Collection, it is used to create Detail Query Sections (see below). If the Navigation is to a single object, its properties are added to the Available Properties list (e.g. BestFriend related properties below). Xalt processes these by using the $expand keyword on a request.



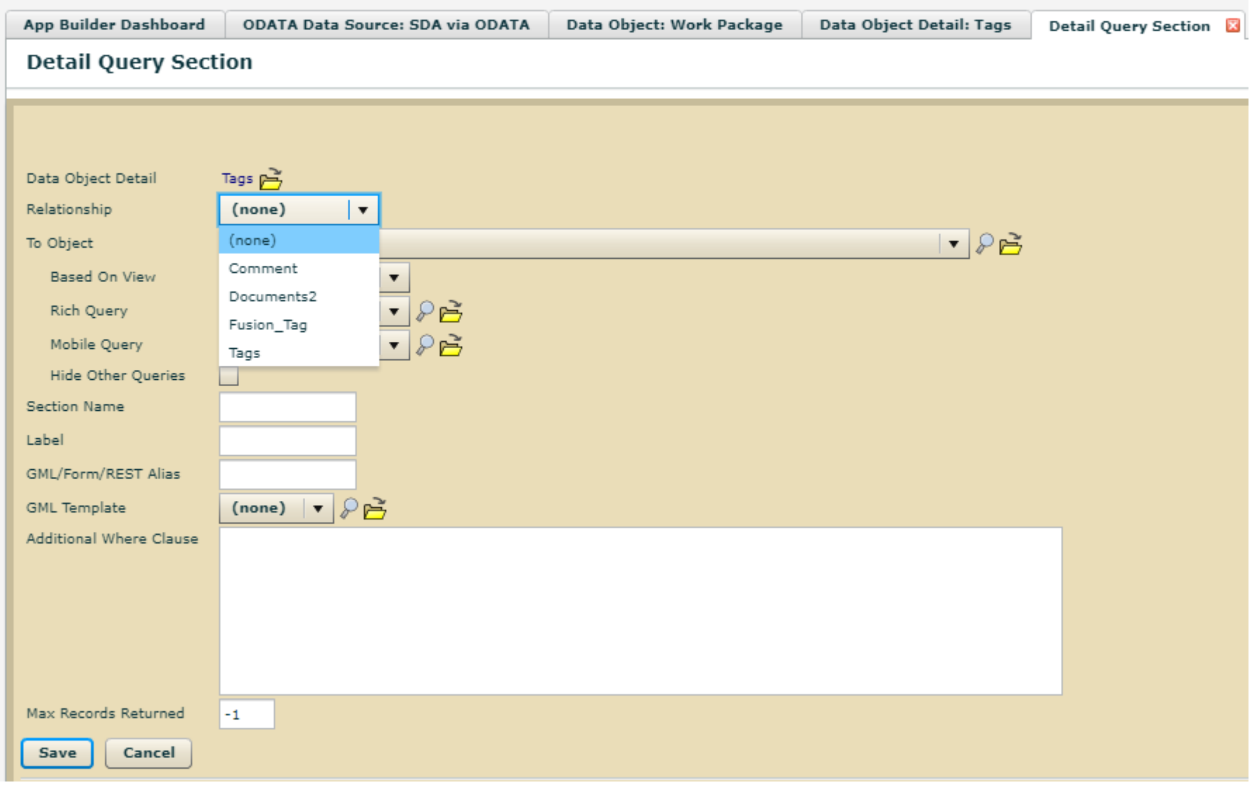
Add Data Object Properties

to Data Object: Person

Available

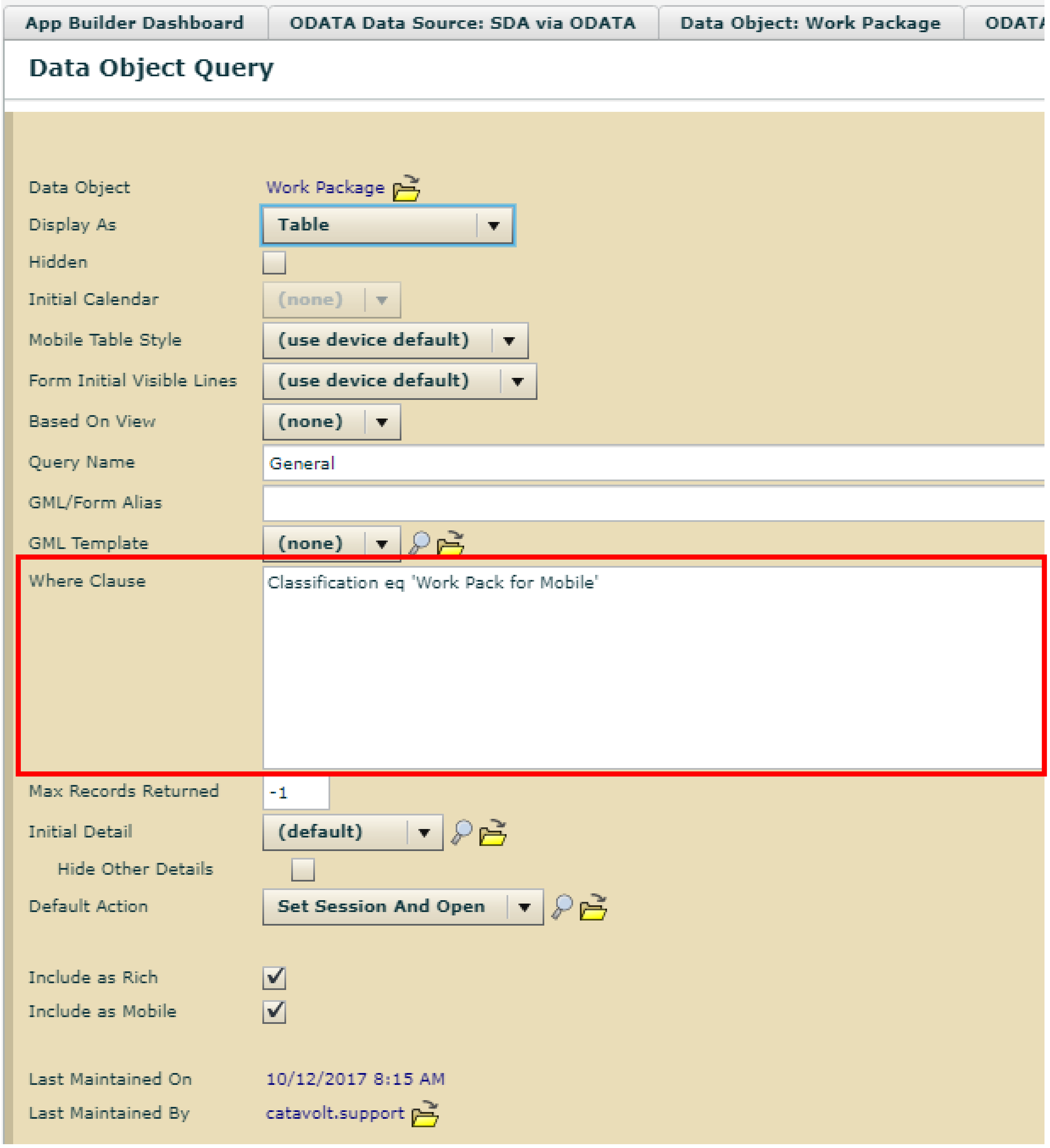
Relationship 1 ▲	Description 2 ▲	Data Type	Name
	Age	Whole Number	Age
	FavoriteFeature	Whole Number	FavoriteFeature
	FirstName	String	FirstName
	Gender	Whole Number	Gender
	HomeAddress/Address	String	HomeAddress/Address
	HomeAddress/City/CountryRegion	String	HomeAddress/City/CountryRegion
	HomeAddress/City/Name	String	HomeAddress/City/Name
	HomeAddress/City/Region	String	HomeAddress/City/Region
	LastName	String	LastName
	MiddleName	String	MiddleName
	UserName	String	UserName
BestFriend	Age	Whole Number	BestFriend/Age
BestFriend	FavoriteFeature	Whole Number	BestFriend/FavoriteFeature
BestFriend	FirstName	String	BestFriend/FirstName
BestFriend	Gender	Whole Number	BestFriend/Gender
BestFriend	HomeAddress/Address	String	BestFriend/HomeAddress/Address
BestFriend	HomeAddress/City/CountryRegion	String	BestFriend/HomeAddress/City/CountryRegion
BestFriend	HomeAddress/City/Name	String	BestFriend/HomeAddress/City/Name
BestFriend	HomeAddress/City/Region	String	BestFriend/HomeAddress/City/Region
BestFriend	LastName	String	BestFriend/LastName
BestFriend	MiddleName	String	BestFriend/MiddleName
BestFriend	UserName	String	BestFriend/UserName

When creating a Detail Query Section (sub-list) for Direct SQL Data Sources, you choose a To Object, which gives you an unconstrained list. You must then fill in an Additional Where Clause in order to restrict the records to just ones for the parent object. The ODATA Data Source extracts all Collection Navigation Properties into the Relationship drop down list (see below). Selecting a Relationship will automatically select the matching To Object and handle the constraining information. You can leave the Additional Where Clause empty or use it to add extra filtering logic.



Where Clauses

Where Clause values are available for ODATA Data Objects. They should use the standard ODATA $filter syntax:



Data Object Query

Data Object	Work Package
Display As	Table
Hidden	□
Initial Calendar	(none)
Mobile Table Style	(use device default)
Form Initial Visible Lines	(use device default)
Based On View	(none)
Query Name	General
GML/Form Alias	
GML Template	(none)
Where Clause	Classification eq 'Work Pack for Mobile'
Max Records Returned	-1
Initial Detail	(default)
Hide Other Details	□
Default Action	Set Session And Open
Include as Rich	☑
Include as Mobile	☑
Last Maintained On	10/12/2017 8:15 AM
Last Maintained By	catavolt.support

Actions

ODATA HTTPActionCall URLs tend to use navigation to identify a resource. For example, to access Task **def** inside Workpackage **abc**, you would use the following standard ODATA URL:

https://base.url.com/Workpackage('abc')/Tasks('def')

The special value ${SELECTED_ODATA_OBJECT_ID} has been added to be able to identify the URL path portion of the selected ODATA object. In the above example for a Task, SELECTED_ODATA_OBJECT_ID would be

Workpackage('abc')/Tasks('def')

If the object is specified as an EntitySet or Singleton inside an EntityContainer, the SELECTED_ODATA_OBJECT_ID will be the direct path to the object (for example, **Tasks('def')**), otherwise it will be the navigation path to the object (for example, **Workpackage('abc')/Tasks('def')**).

We also allow a special function ${SELECTED_ODATA_OBJECT_ID-x} to be able to specify URLs that are rooted in parent paths of the ODATA Object ID path. For example, if the Object ID for an ODATA Data Object is:

Sites('HEX')/PlantGroups('DEMO')/Projects('0')/Instrument('4293')

${SELECTED_ODATA_OBJECT_ID} = Sites('HEX')/PlantGroups('DEMO')/Projects('0')/Instrument ('4293')
${SELECTED_ODATA_OBJECT_ID-1} = Sites('HEX')/PlantGroups('DEMO')/Projects('0')
${SELECTED_ODATA_OBJECT_ID-2} = Sites('HEX')/PlantGroups('DEMO')
${SELECTED_ODATA_OBJECT_ID-3} = Sites('HEX')

and so on.

Completion Actions

When creating an Action, you can specify a Completion Action of Open Object or Open List and specify a Where Clause to identify the record(s) to display. In an ODATA Data Source, Xalt will perform a direct (top level) request with a $filter in order to resolve the records. Unfortunately, this does not work for Data Objects that are not in an EntitySet (i.e. they are not "top-level" in the ODATA document). For example, in ODATA you may have Person and Trip EntityTypes. Person is in the EntitySet, meaning you can directly request a list of People (http://<base url>/Persons). Trip is not in the EntitySet but is a NavigationProperty off of Person. This means that you cannot directly request a list of Trips (http://<base url>/Trips), but instead must go through a Person (**Error! Hyperlink reference not valid.** url>/Persons('BobBurdock')/Trips).

Xalt allows you to directly specify the URL Path to query a List or Object in the Where Clause using two special keywords:

1. *OBJECT_PATH – If you are doing an Open Object, you can use this keyword to specify the direct path to access the object. In our example above, you would use *OBJECT_PATH(Persons('BobBurdock')/Trips(0) in the Where Clause
2. *LIST_PATH – If you are doing an Open List, you can use this keyword to specify the direct path to access the list. In our example above, you would use *OBJECT_PATH(Persons('BobBurdock')/Trips in the Where Clause

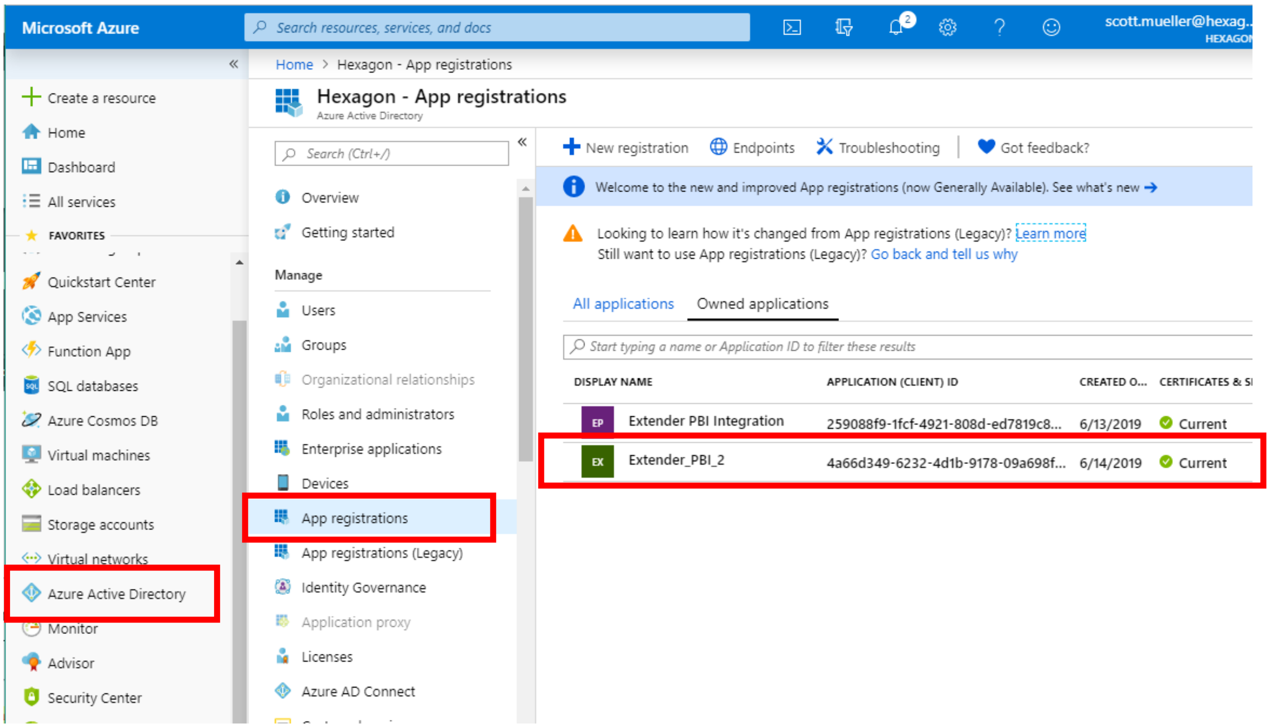
Appendix F: Microsoft Power BI Integration

Finding ID values

When creating a Power BI Data Source you need to enter the following values from Power BI and Azure AD:

- Tenant ID
- Application (Client ID)
- Client Secret
- Group (Workspace) ID

The first 3 values can be found in Azure AD (https://portal.azure.com). Go to Azure Active Directory and click on App registrations, then select your application:

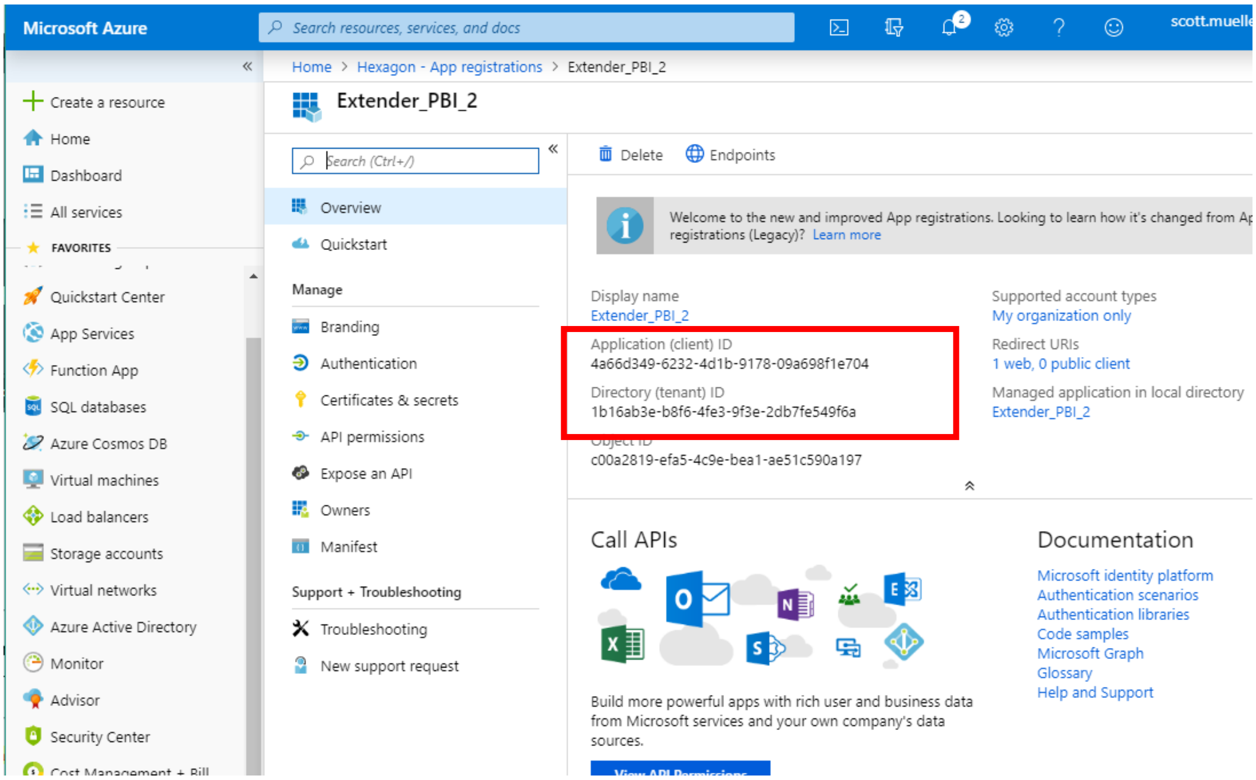


Welcome to the new and improved App registrations (now Generally Available). See what's new →

Looking to learn how it's changed from App registrations (Legacy)? Learn more
Still want to use App registrations (Legacy)? Go back and tell us why

DISPLAY NAME	APPLICATION (CLIENT) ID	CREATED O...	CERTIFICATES & SI
EP Extender PBI Integration	259088f9-1fcf-4921-808d-ed7819c8...	6/13/2019	✓ Current
EX Extender_PBI_2	4a66d349-6232-4d1b-9178-09a698f...	6/14/2019	✓ Current

From here you can find the Application (Client) ID and Tenant ID:



Welcome to the new and improved App registrations. Looking to learn how it's changed from Ap registrations (Legacy)? Learn more

Display name
Extender_PBI_2

Application (client) ID
4a66d349-6232-4d1b-9178-09a698f1e704

Directory (tenant) ID
1b16ab3e-b8f6-4fe3-9f3e-2db7fe549f6a

Object ID
c00a2819-efa5-4c9e-bea1-ae51c590a197

Supported account types
My organization only

Redirect URIs
1 web, 0 public client

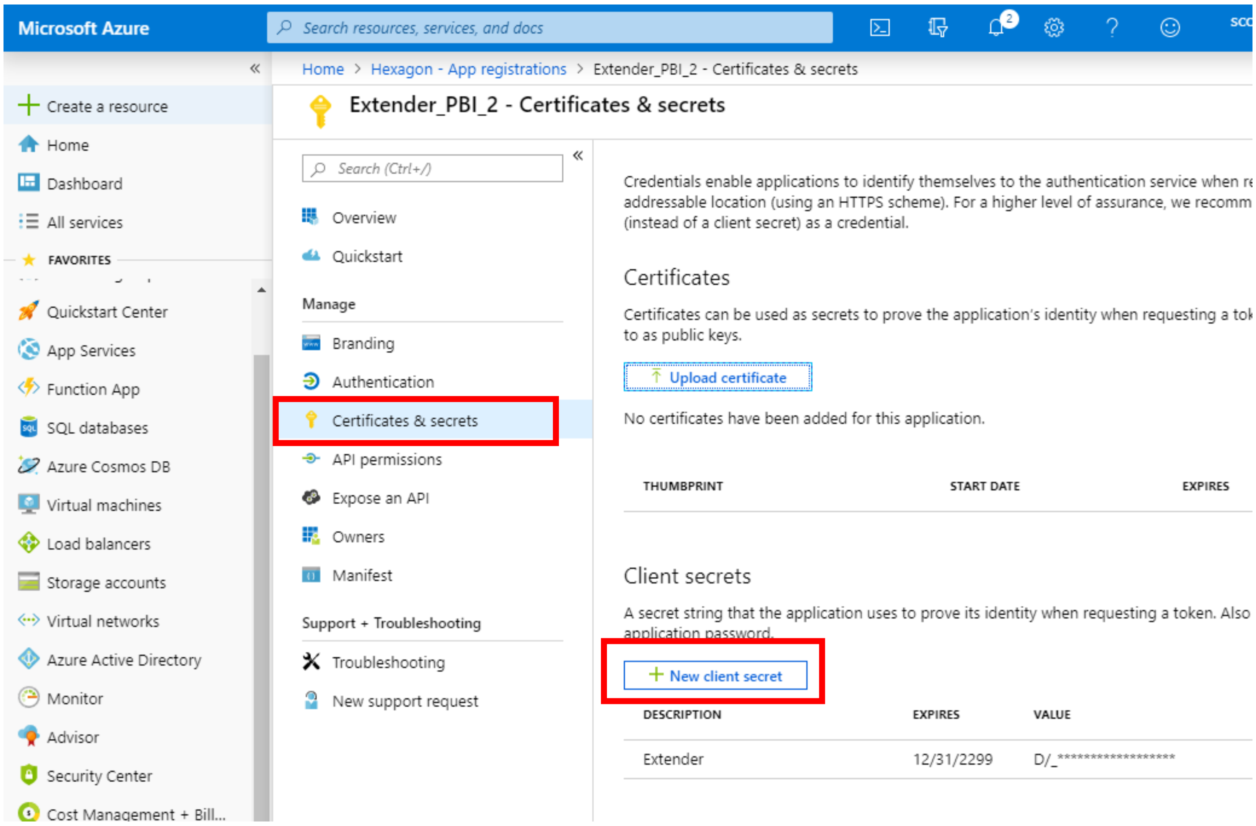
Managed application in local directory
Extender_PBI_2

Call APIs

Build more powerful apps with rich user and business data from Microsoft services and your own company's data sources.

Documentation

You can then click on the “Certificates and secrets” section to generate a new Client Secret if you do not remember the current secret:



Credentials enable applications to identify themselves to the authentication service when re addressable location (using an HTTPS scheme). For a higher level of assurance, we recomm (instead of a client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a tok to as public keys.

No certificates have been added for this application.

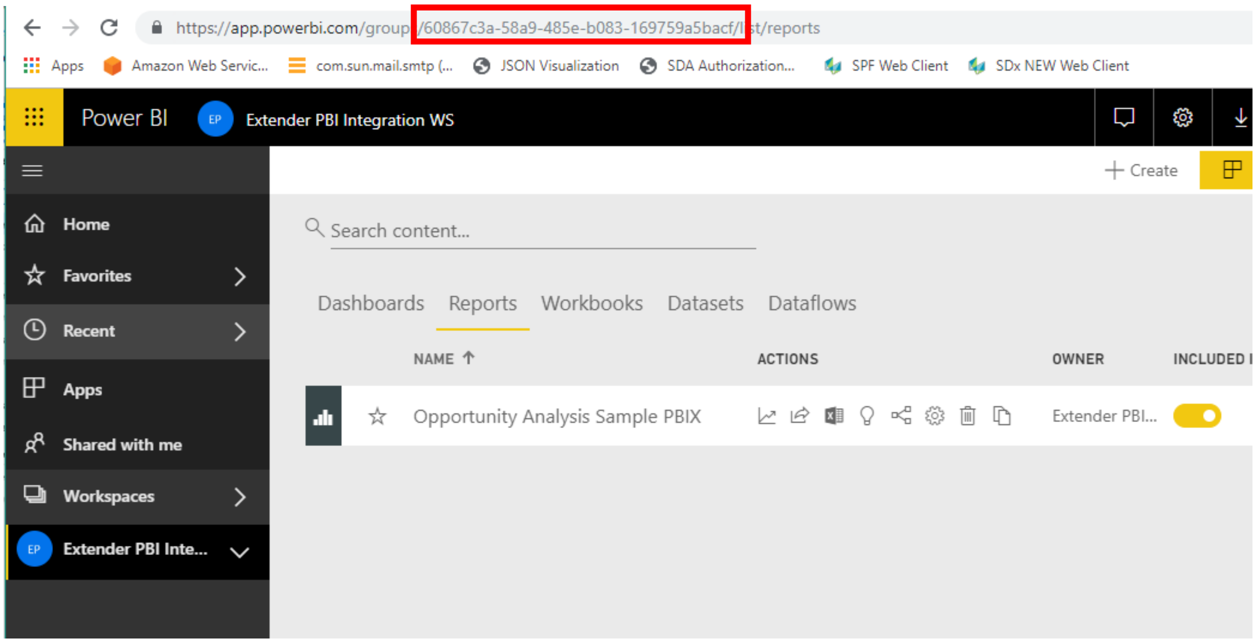
THUMBPRINT	START DATE	EXPIRES

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also application password.

DESCRIPTION	EXPIRES	VALUE
Extender	12/31/2299	D/_******************

To find the Group (Workspace) ID, connect to Power BI (https://app.powerbi.com) and select your Workspace. The Group (Workspace) ID will be the token value in the URL between “groups/” and “/list”:



https://app.powerbi.com/groups/60867c3a-58a9-485e-b083-169759a5bacf/list/reports

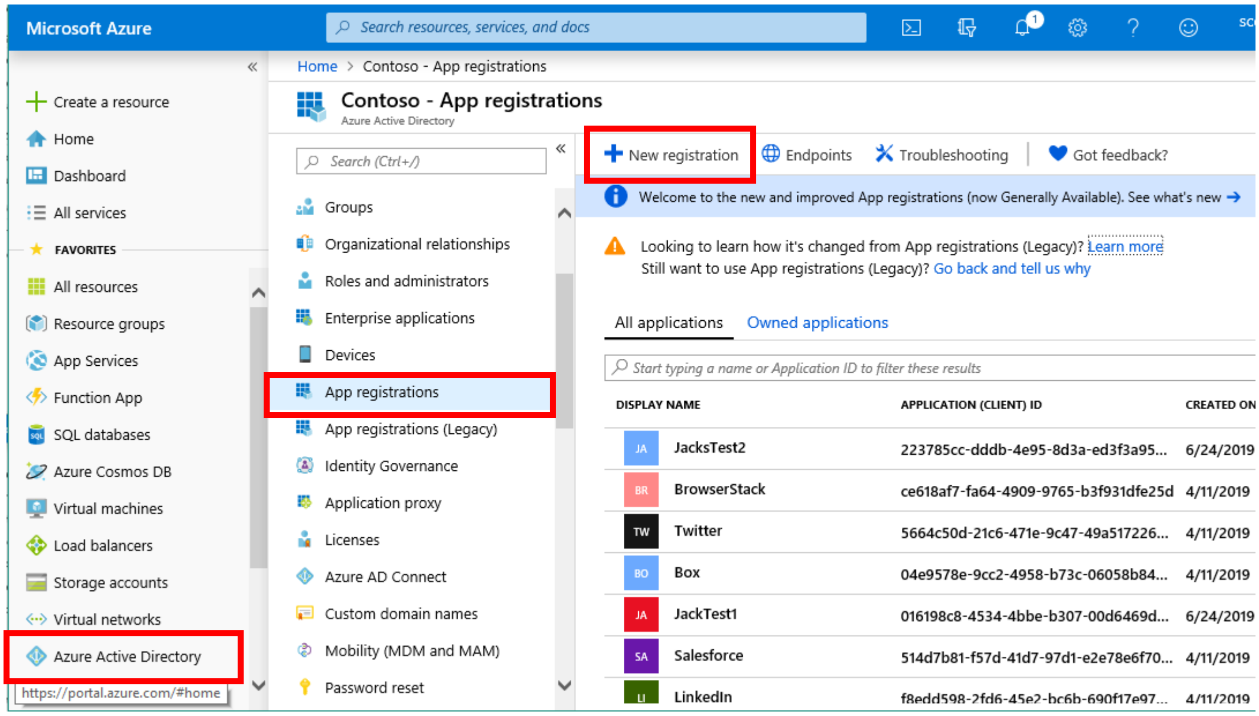
NAME ↑	ACTIONS	OWNER	INCLUDED I
Opportunity Analysis Sample PBIX		Extender PBI...	

Embedded Power BI Applications

In order to embed Power BI into client applications, you use Azure AD to create an Application over the Power BI Workspace. There are two approaches that can be used to do this, the "User-Owns-Data" and "App-Owns-Data" approach. In the "User-Owns-Data" approach, a "master" user is used that manages access to the application. In the "App-Owns-Data" approach, a service principal is created over the application that manages access. Of the two approaches, Microsoft recommends using the "App-Owns-Data" approach as a best practice. We have added support for both approaches in Xalt. We assume that you have already created a Power BI Workspace when following the instructions below

App-Owns-Data

Go to Azure AD (https://portal.azure.com). Go to Azure Active Directory and click on App registrations, then select New Registration:

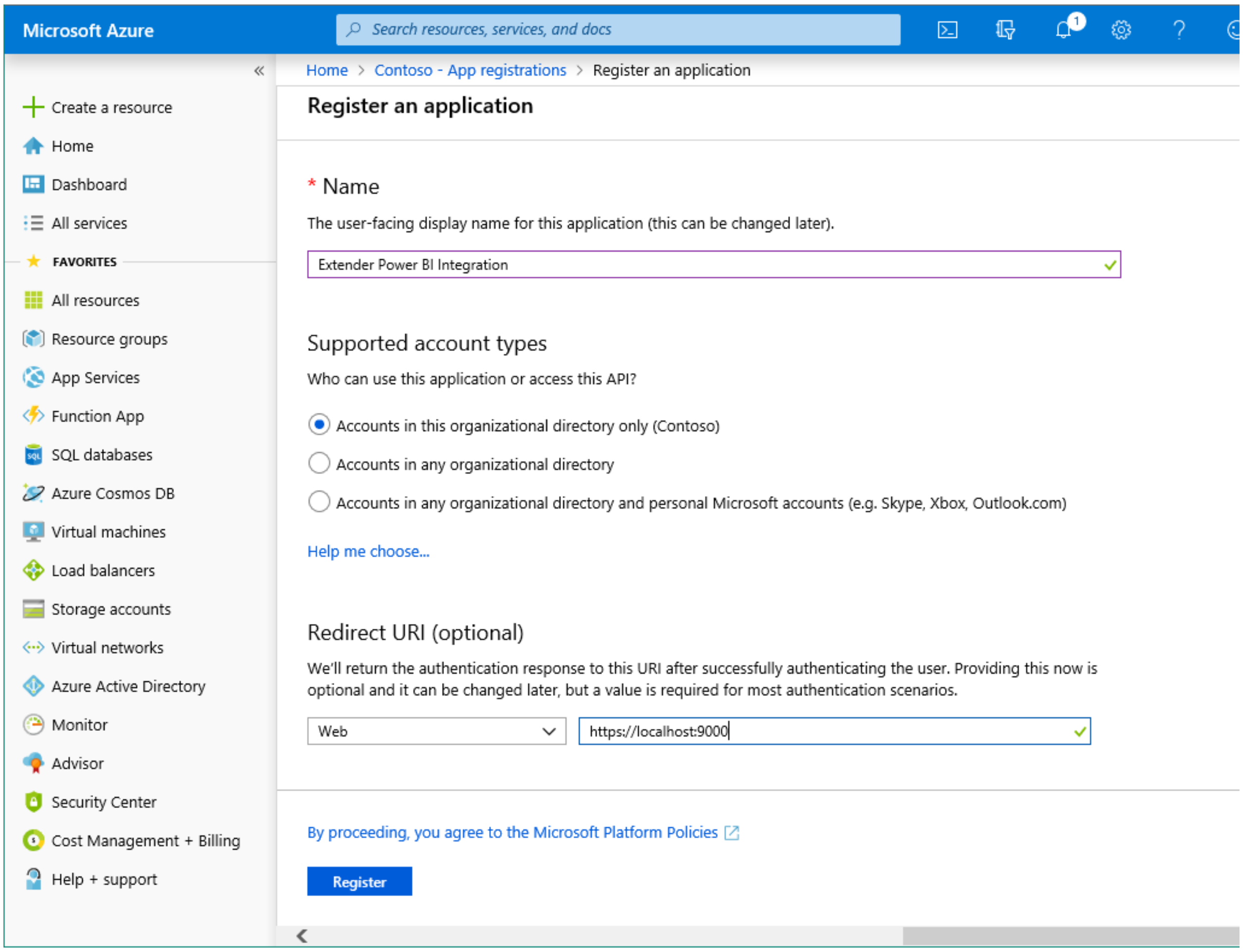


Welcome to the new and improved App registrations (now Generally Available). See what's new →

Looking to learn how it's changed from App registrations (Legacy)? Learn more
Still want to use App registrations (Legacy)? Go back and tell us why

DISPLAY NAME	APPLICATION (CLIENT) ID	CREATED ON
JA JacksTest2	223785cc-dddb-4e95-8d3a-ed3f3a95...	6/24/2019
BR BrowserStack	ce618af7-fa64-4909-9765-b3f931dfe25d	4/11/2019
TW Twitter	5664c50d-21c6-471e-9c47-49a517226...	4/11/2019
BO Box	04e9578e-9cc2-4958-b73c-06058b84...	4/11/2019
JA JackTest1	016198c8-4534-4bbe-b307-00d6469d...	6/24/2019
SA Salesforce	514d7b81-f57d-41d7-97d1-e2e78e6f70...	4/11/2019
LI LinkedIn	f8edd598-2fd6-45e2-bc6b-690f17e97...	4/11/2019

Enter the application display name and Redirect URI. Note that you may enter any value for Redirect URI, as it will not be referenced by the Xalt application.



Register an application

* Name

The user-facing display name for this application (this can be changed later).

Extender Power BI Integration ✓

Supported account types

Who can use this application or access this API?

● Accounts in this organizational directory only (Contoso)

○ Accounts in any organizational directory

○ Accounts in any organizational directory and personal Microsoft accounts (e.g. Skype, Xbox, Outlook.com)

Help me choose...

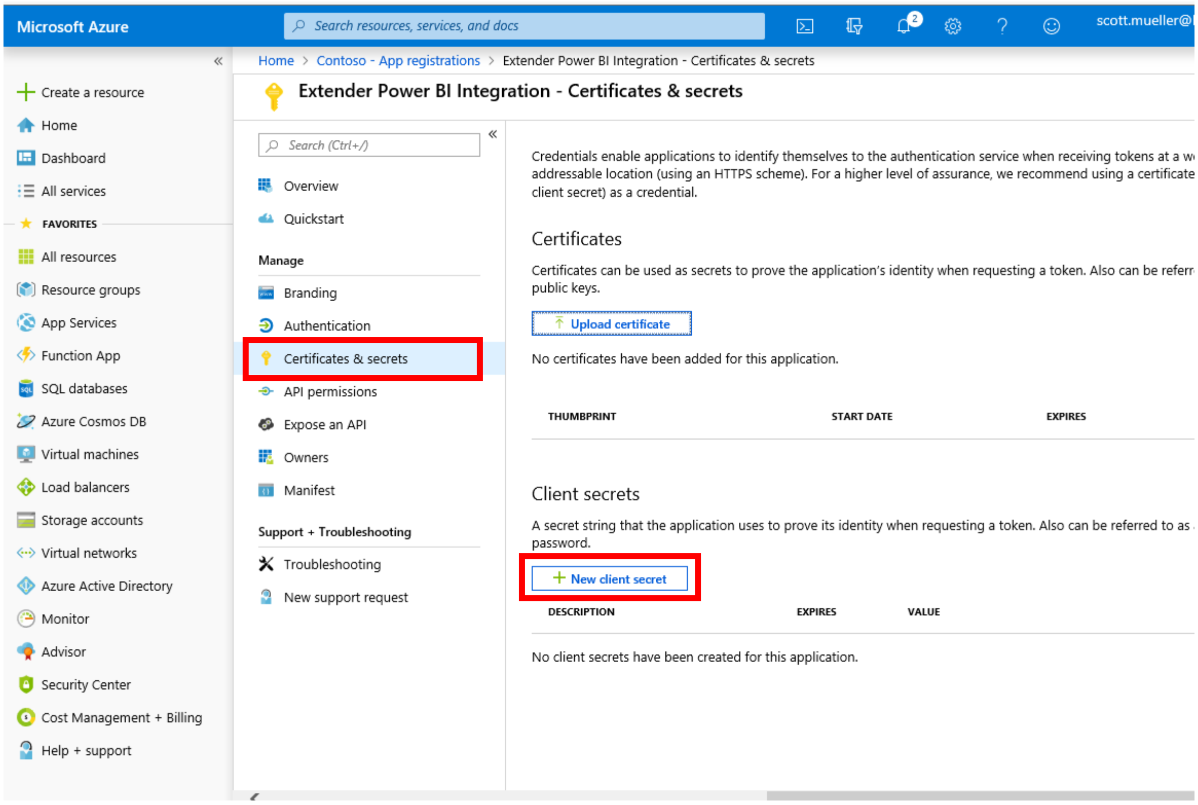
Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web | https://localhost:9000 ✓

By proceeding, you agree to the Microsoft Platform Policies

Next, go to Certificates and secrets and select New client secret:



Credentials enable applications to identify themselves to the authentication service when receiving tokens at a w addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referr public keys.

No certificates have been added for this application.

THUMBPRINT	START DATE	EXPIRES

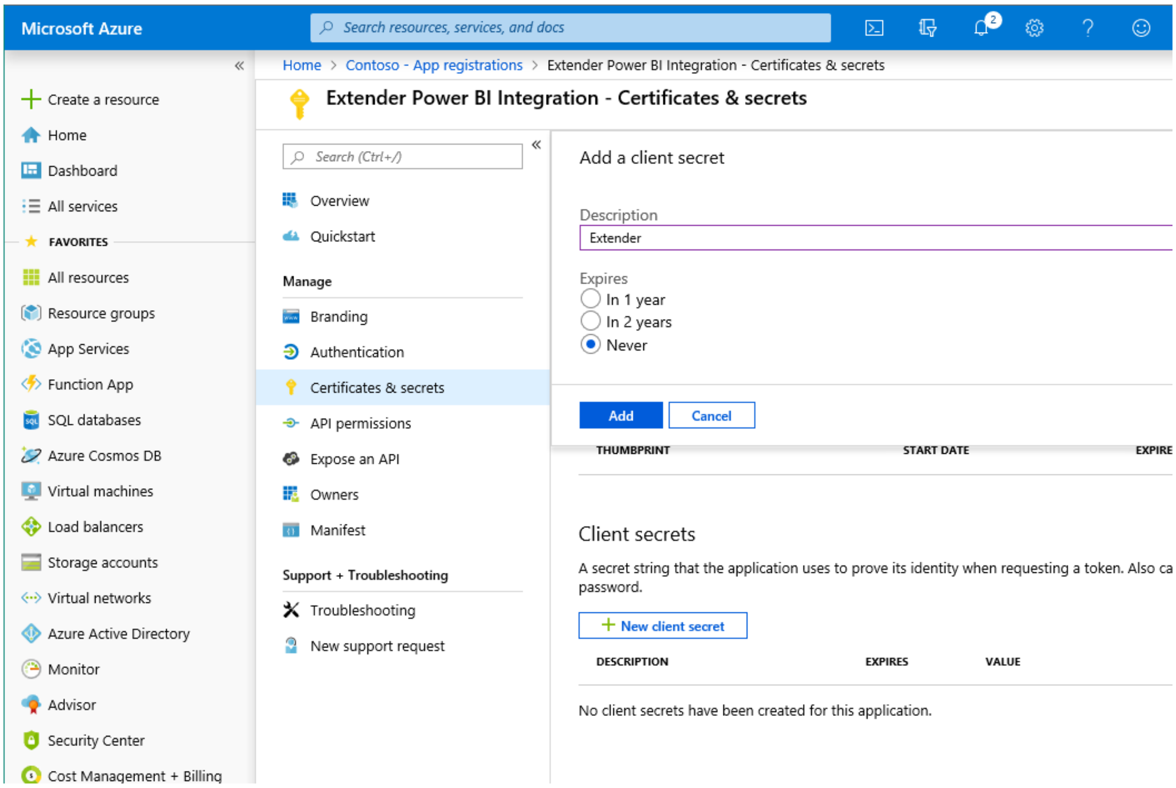
Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as password.

DESCRIPTION	EXPIRES	VALUE

No client secrets have been created for this application.

Enter the information to add a client secret. You may set an Expires value of your choosing:



Add a client secret

Description: Extender

Expires
○ In 1 year
○ In 2 years
● Never

THUMBPRINT	START DATE	EXPIRE

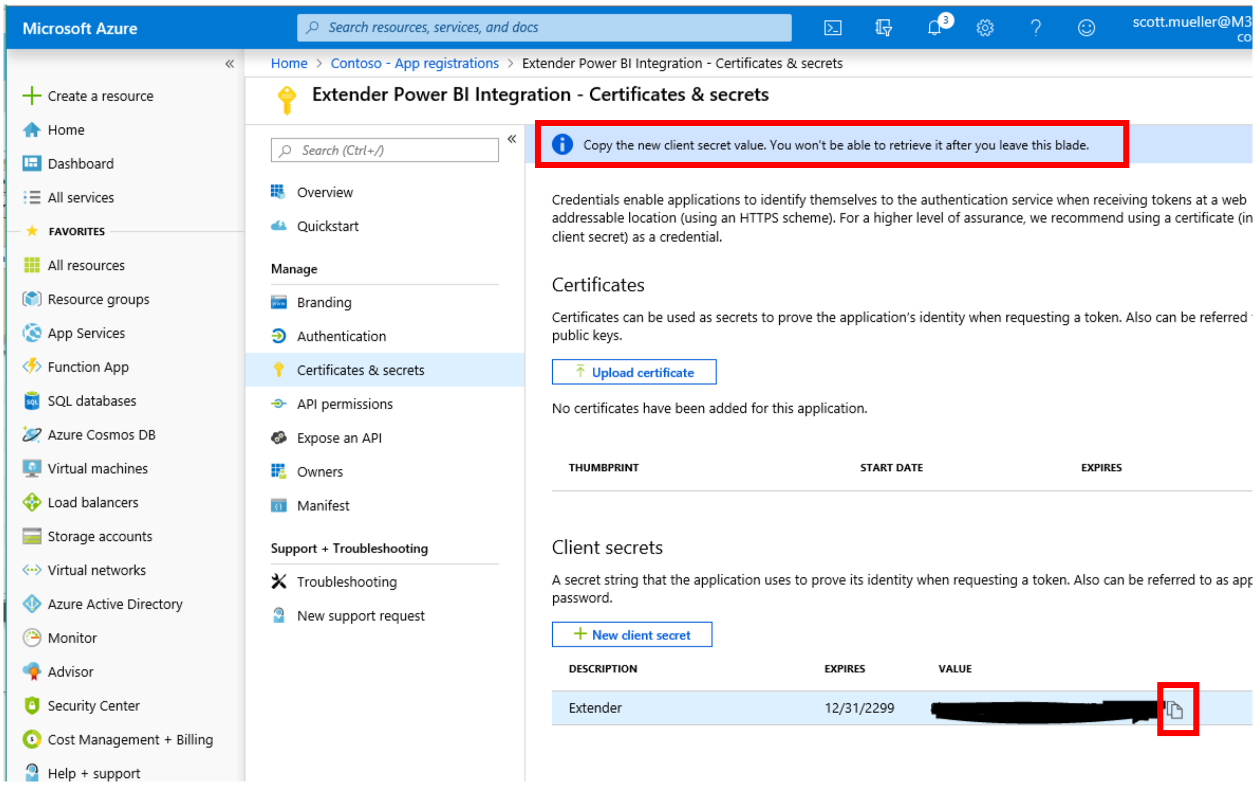
Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also ca password.

DESCRIPTION	EXPIRES	VALUE

No client secrets have been created for this application.

Make sure to copy the value, as you cannot retrieve it again after it has been created:



Extender Power BI Integration - Certificates & secrets

Copy the new client secret value. You won't be able to retrieve it after you leave this blade.

Credentials enable applications to identify themselves to the authentication service when receiving tokens at a web addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certificate (in client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be referred public keys.

No certificates have been added for this application.

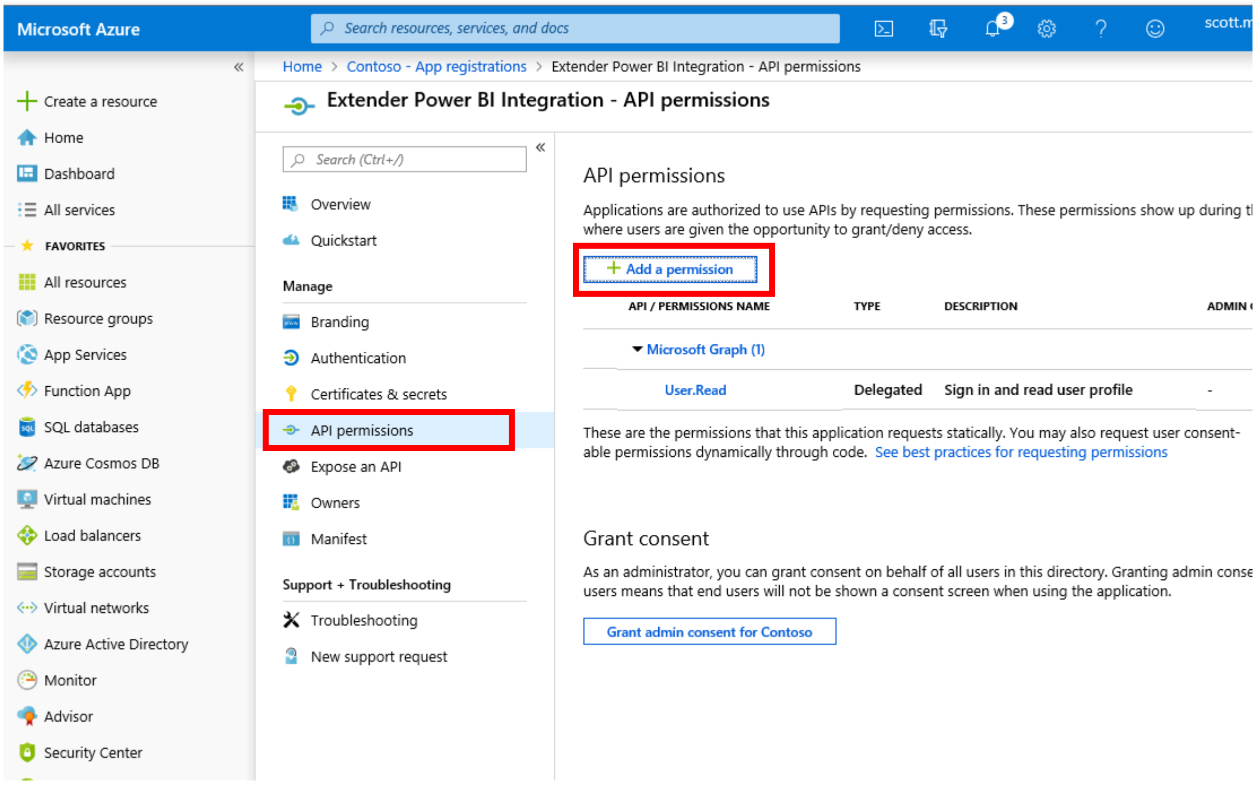
THUMBPRINT	START DATE	EXPIRES

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to as app password.

DESCRIPTION	EXPIRES	VALUE
Extender	12/31/2299	[redacted]

Next, go to API Permissions and choose Add a permission:



Extender Power BI Integration - API permissions

API permissions

Applications are authorized to use APIs by requesting permissions. These permissions show up during t where users are given the opportunity to grant/deny access.

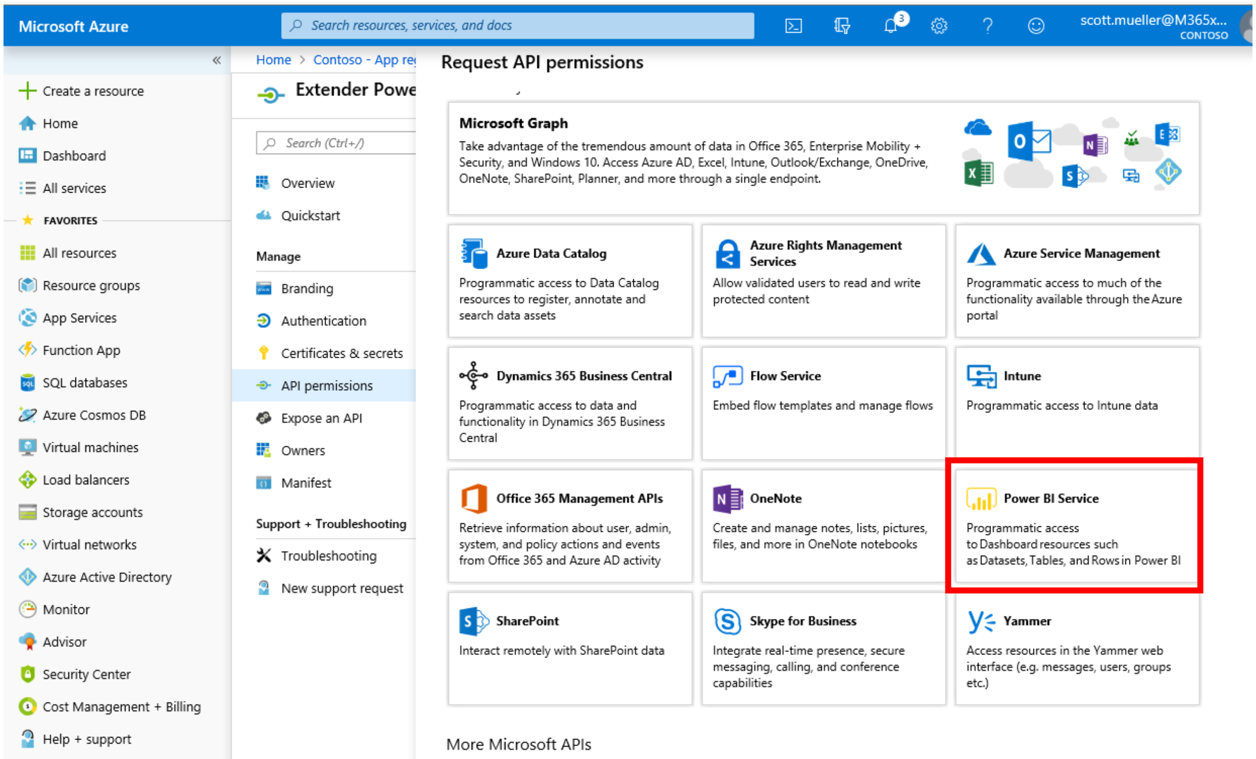
API / PERMISSIONS NAME	TYPE	DESCRIPTION	ADMIN (
▾ Microsoft Graph (1)			
User.Read	Delegated	Sign in and read user profile	-

These are the permissions that this application requests statically. You may also request user consent-able permissions dynamically through code. See best practices for requesting permissions

Grant consent

As an administrator, you can grant consent on behalf of all users in this directory. Granting admin conse users means that end users will not be shown a consent screen when using the application.

Select Power BI Service:



Request API permissions

Microsoft Graph
Take advantage of the tremendous amount of data in Office 365, Enterprise Mobility + Security, and Windows 10. Access Azure AD, Excel, Intune, Outlook/Exchange, OneDrive, OneNote, SharePoint, Planner, and more through a single endpoint.

Azure Data Catalog
Programmatic access to Data Catalog resources to register, annotate and search data assets

Azure Rights Management Services
Allow validated users to read and write protected content

Azure Service Management
Programmatic access to much of the functionality available through the Azure portal

Dynamics 365 Business Central
Programmatic access to data and functionality in Dynamics 365 Business Central

Flow Service
Embed flow templates and manage flows

Intune
Programmatic access to Intune data

Office 365 Management APIs
Retrieve information about user, admin, system, and policy actions and events from Office 365 and Azure AD activity

OneNote
Create and manage notes, lists, pictures, files, and more in OneNote notebooks

Power BI Service
Programmatic access to Dashboard resources such as Datasets, Tables, and Rows in Power BI

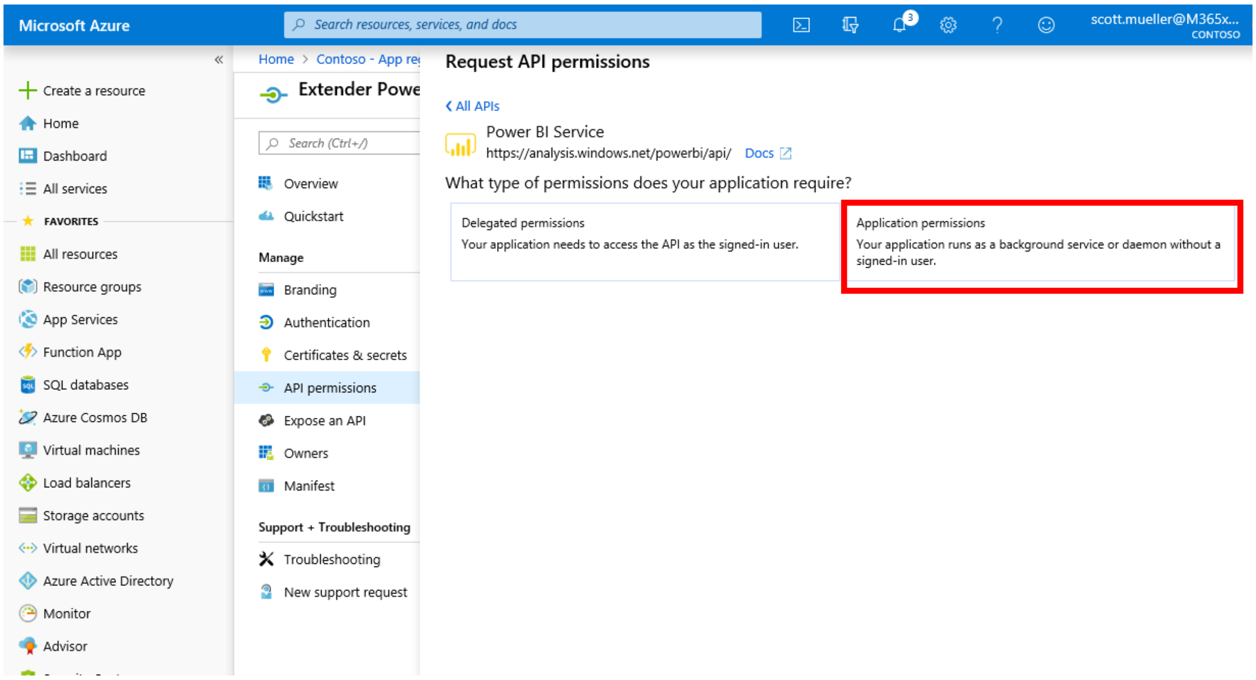
SharePoint
Interact remotely with SharePoint data

Skype for Business
Integrate real-time presence, secure messaging, calling, and conference capabilities

Yammer
Access resources in the Yammer web interface (e.g. messages, users, groups etc.)

More Microsoft APIs

Select Application permissions:



Request API permissions

< All APIs

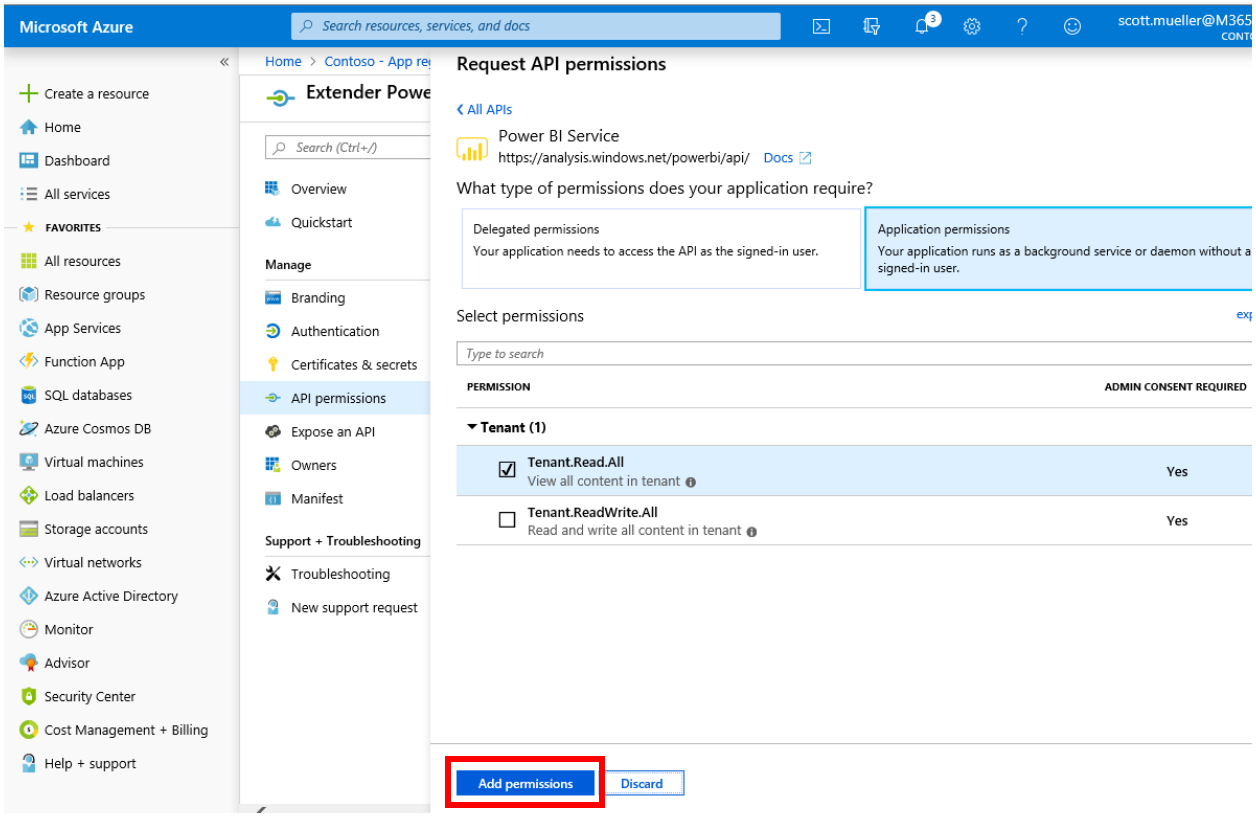
Power BI Service
https://analysis.windows.net/powerbi/api/ Docs

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

Application permissions
Your application runs as a background service or daemon without a signed-in user.

Select Tenant.Read.All permissions:



Request API permissions

< All APIs

Power BI Service
https://analysis.windows.net/powerbi/api/ Docs

What type of permissions does your application require?

Delegated permissions
Your application needs to access the API as the signed-in user.

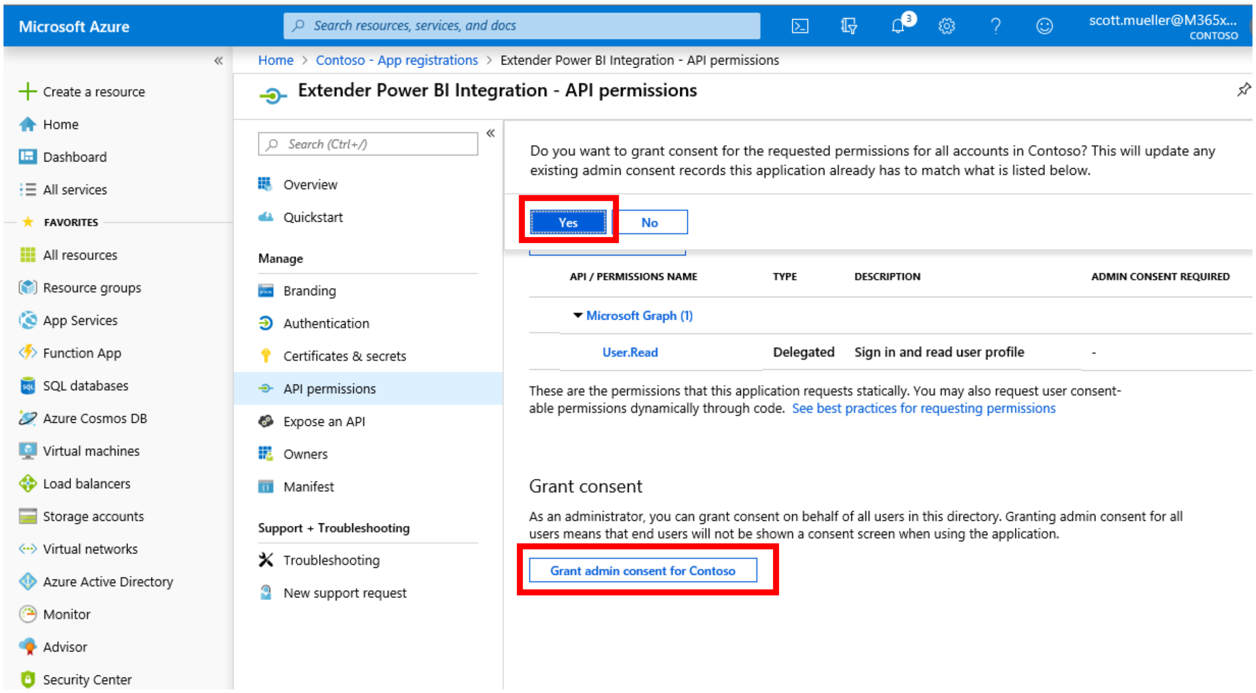
Application permissions
Your application runs as a background service or daemon without a signed-in user.

Select permissions

Type to search

PERMISSION	ADMIN CONSENT REQUIRED
▾ Tenant (1)	
☑ Tenant.Read.All View all content in tenant	Yes
☐ Tenant.ReadWrite.All Read and write all content in tenant	Yes

Note that you get a message saying consent is required. Select the "Grant admin consent for …" button to grant consent:



Extender Power BI Integration - API permissions

Do you want to grant consent for the requested permissions for all accounts in Contoso? This will update any existing admin consent records this application already has to match what is listed below.

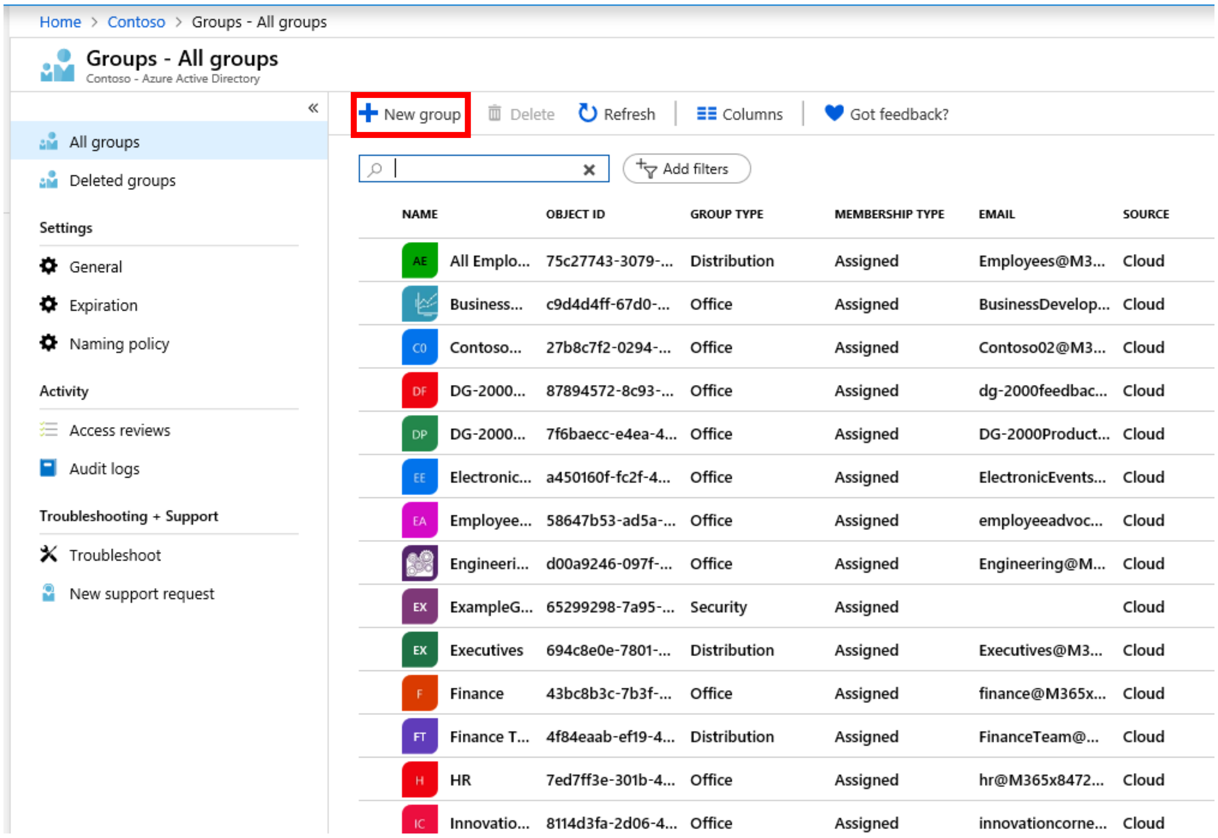
API / PERMISSIONS NAME	TYPE	DESCRIPTION	ADMIN CONSENT REQUIRED
▾ Microsoft Graph (1)			
User.Read	Delegated	Sign in and read user profile	-

These are the permissions that this application requests statically. You may also request user consent-able permissions dynamically through code. See best practices for requesting permissions

Grant consent

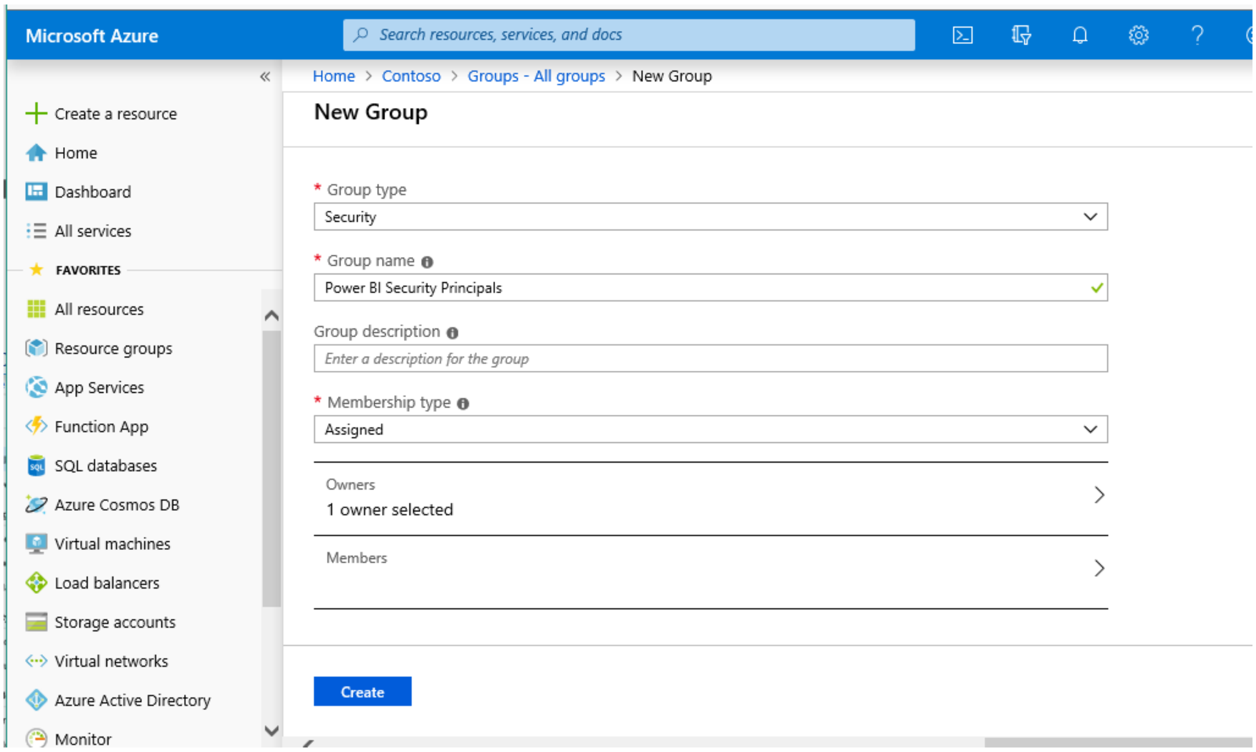
As an administrator, you can grant consent on behalf of all users in this directory. Granting admin consent for all users means that end users will not be shown a consent screen when using the application.

Next, go to Azure Active Directory, Groups, and select New group:



	NAME	OBJECT ID	GROUP TYPE	MEMBERSHIP TYPE	EMAIL	SOURCE
AE	All Emplo...	75c27743-3079-...	Distribution	Assigned	Employees@M3...	Cloud
	Business...	c9d4d4ff-67d0-...	Office	Assigned	BusinessDevelop...	Cloud
CO	Contoso...	27b8c7f2-0294-...	Office	Assigned	Contoso02@M3...	Cloud
DF	DG-2000...	87894572-8c93-...	Office	Assigned	dg-2000feedbac...	Cloud
DP	DG-2000...	7f6baecc-e4ea-4...	Office	Assigned	DG-2000Product...	Cloud
EE	Electronic...	a450160f-fc2f-4...	Office	Assigned	ElectronicEvents...	Cloud
EA	Employee...	58647b53-ad5a-...	Office	Assigned	employeeadvoc...	Cloud
	Engineeri...	d00a9246-097f-...	Office	Assigned	Engineering@M...	Cloud
EX	ExampleG...	65299298-7a95-...	Security	Assigned		Cloud
EX	Executives	694c8e0e-7801-...	Distribution	Assigned	Executives@M3...	Cloud
F	Finance	43bc8b3c-7b3f-...	Office	Assigned	finance@M365x...	Cloud
FT	Finance T...	4f84eaab-ef19-4...	Distribution	Assigned	FinanceTeam@...	Cloud
H	HR	7ed7ff3e-301b-4...	Office	Assigned	hr@M365x8472...	Cloud
IC	Innovatio...	8114d3fa-2d06-4...	Office	Assigned	innovationcorne...	Cloud

Create a new Security Group to hold the Application's Service Principal:



New Group

* Group type: Security

* Group name: Power BI Security Principals

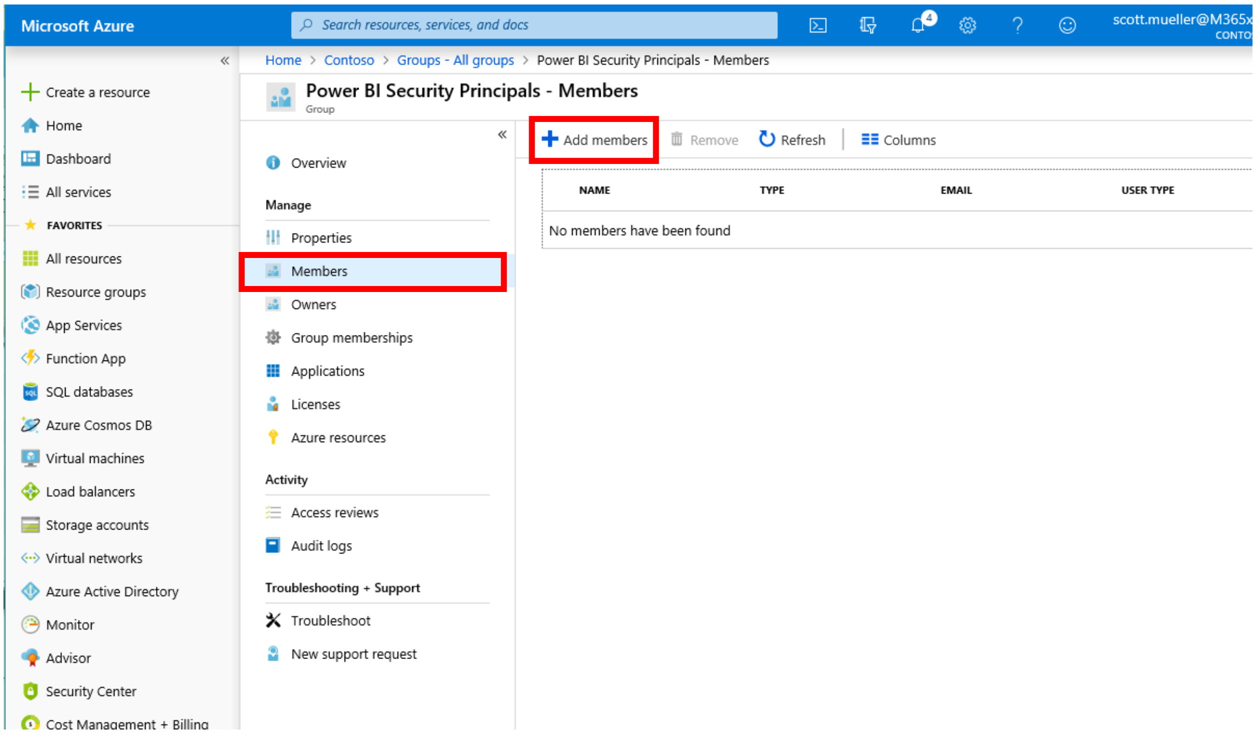
Group description: Enter a description for the group

* Membership type: Assigned

Owners: 1 owner selected

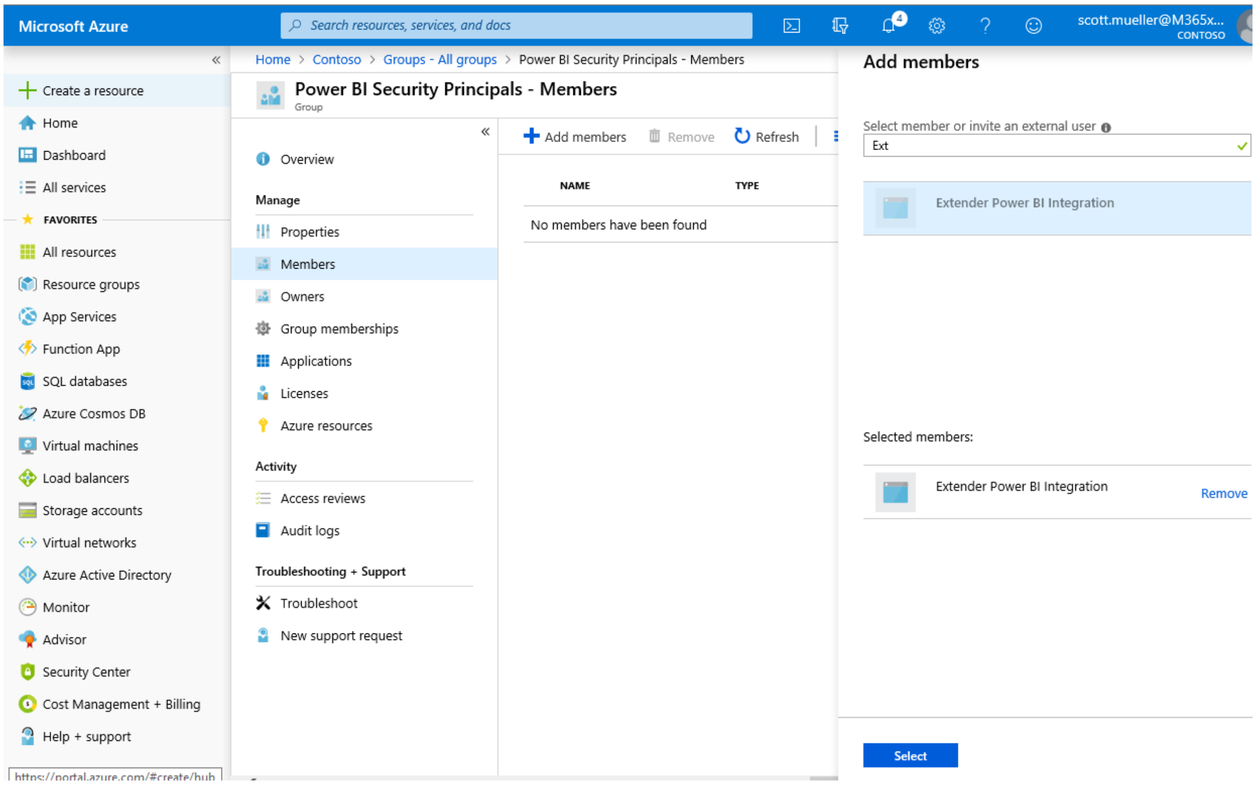
Members

Next, go to Members and select Add members:



No members have been found

Type in the name of the Application created above and add it to the member list



No members have been found

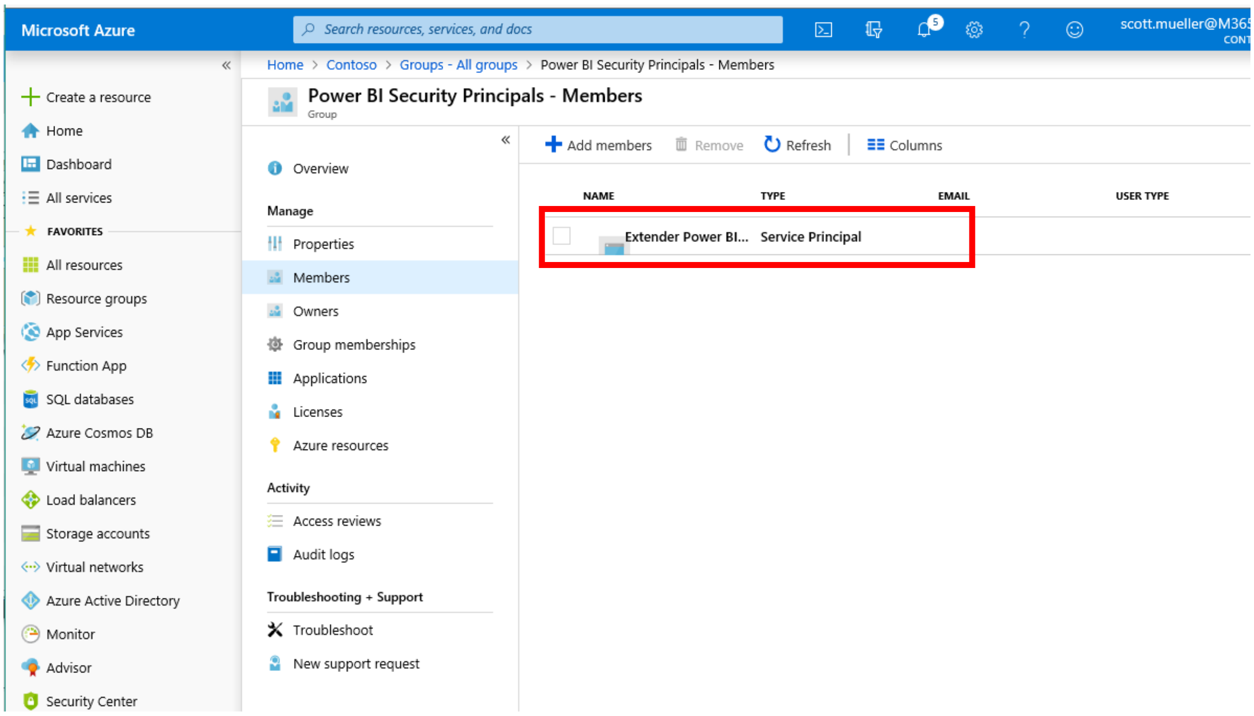
Select member or invite an external user: Ext

Extender Power BI Integration

Selected members:

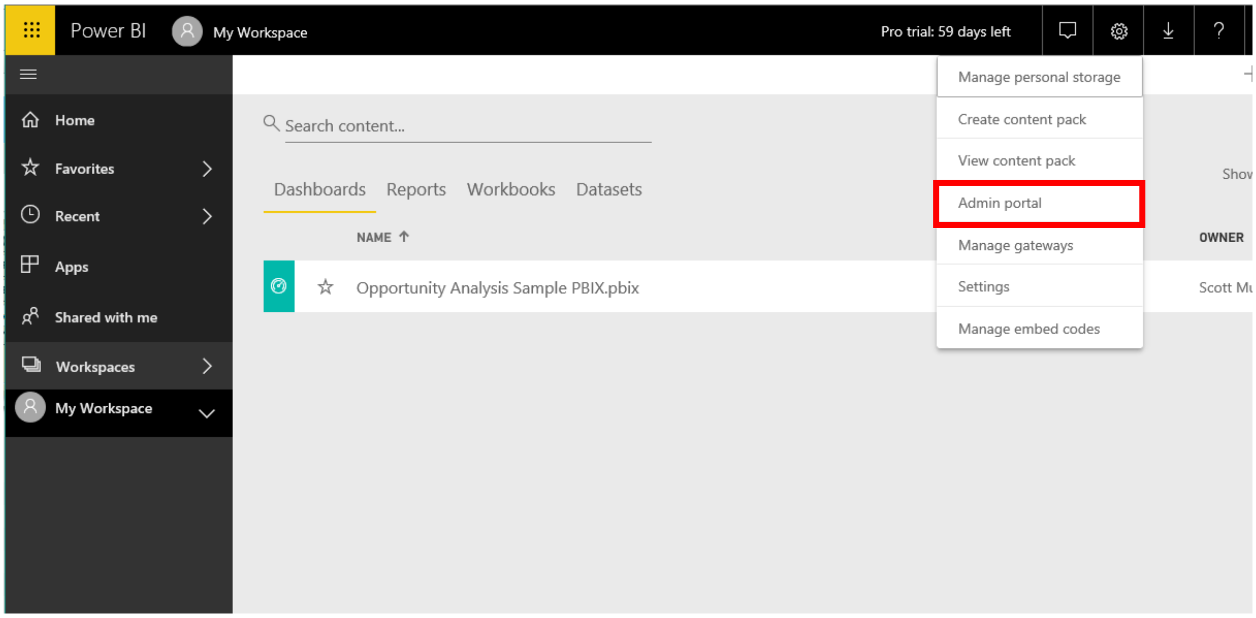
Extender Power BI Integration

You should now see the Application with the type of Service Principal as a group member:

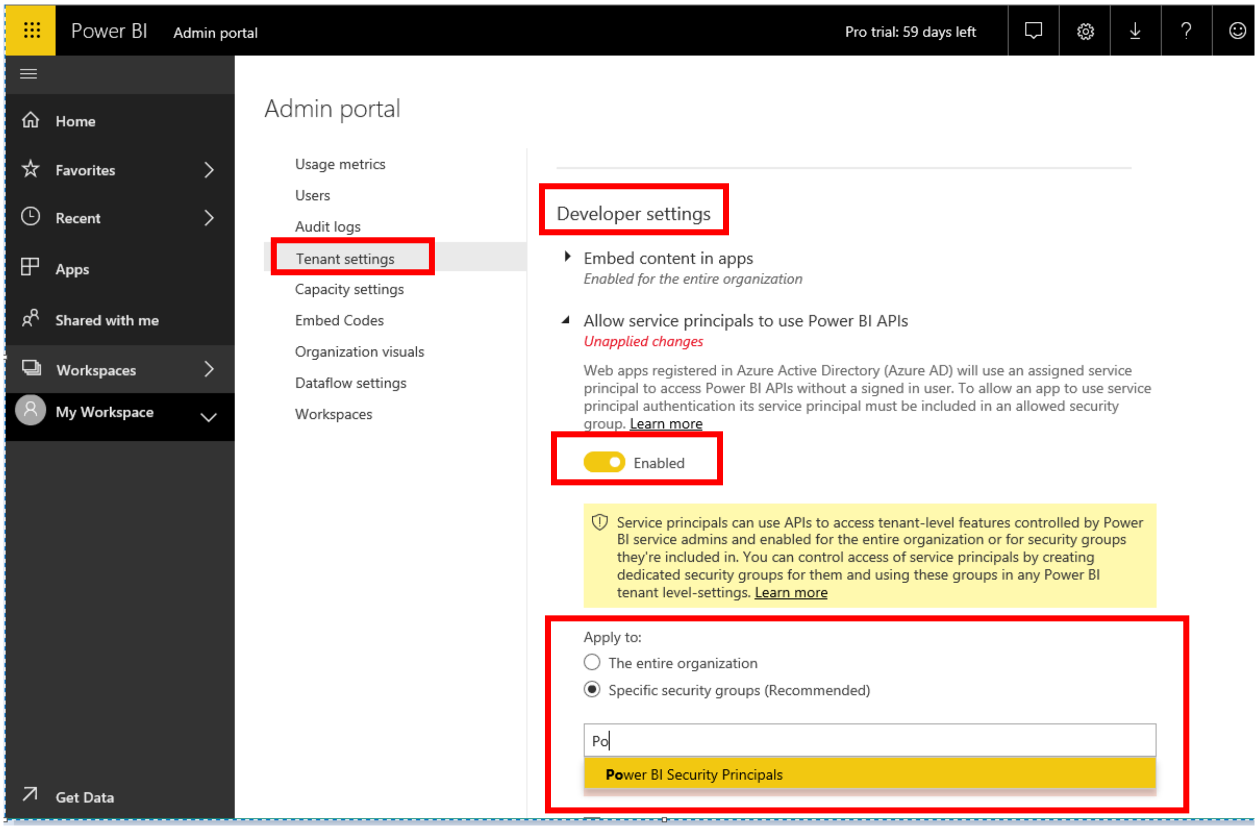


NAME	TYPE	EMAIL	USER TYPE
Extender Power BI...	Service Principal		

Now, connect to Power BI (https://app.powerbi.com) and enter the Admin portal:



Go to Tenant Settings, Developer settings. Verify that Allow service principals to use Power BI APIs is set to Enabled. Under Specific security groups, add the name of the Security Group you created above:



Admin portal

Developer settings

Embed content in apps
Enabled for the entire organization

Allow service principals to use Power BI APIs
Unapplied changes

Web apps registered in Azure Active Directory (Azure AD) will use an assigned service principal to access Power BI APIs without a signed in user. To allow an app to use service principal authentication its service principal must be included in an allowed security group. Learn more

Enabled

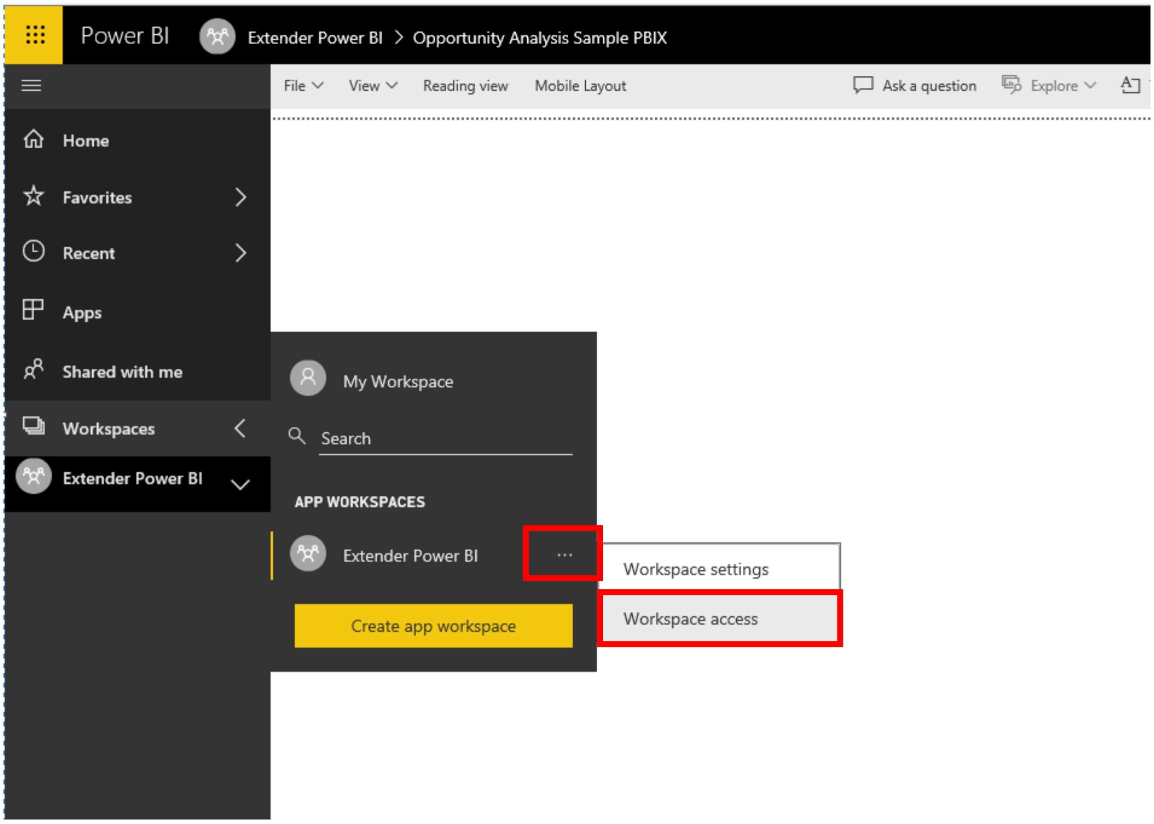
Service principals can use APIs to access tenant-level features controlled by Power BI service admins and enabled for the entire organization or for security groups they're included in. You can control access of service principals by creating dedicated security groups for them and using these groups in any Power BI tenant level-settings. Learn more

Apply to:
○ The entire organization
● Specific security groups (Recommended)

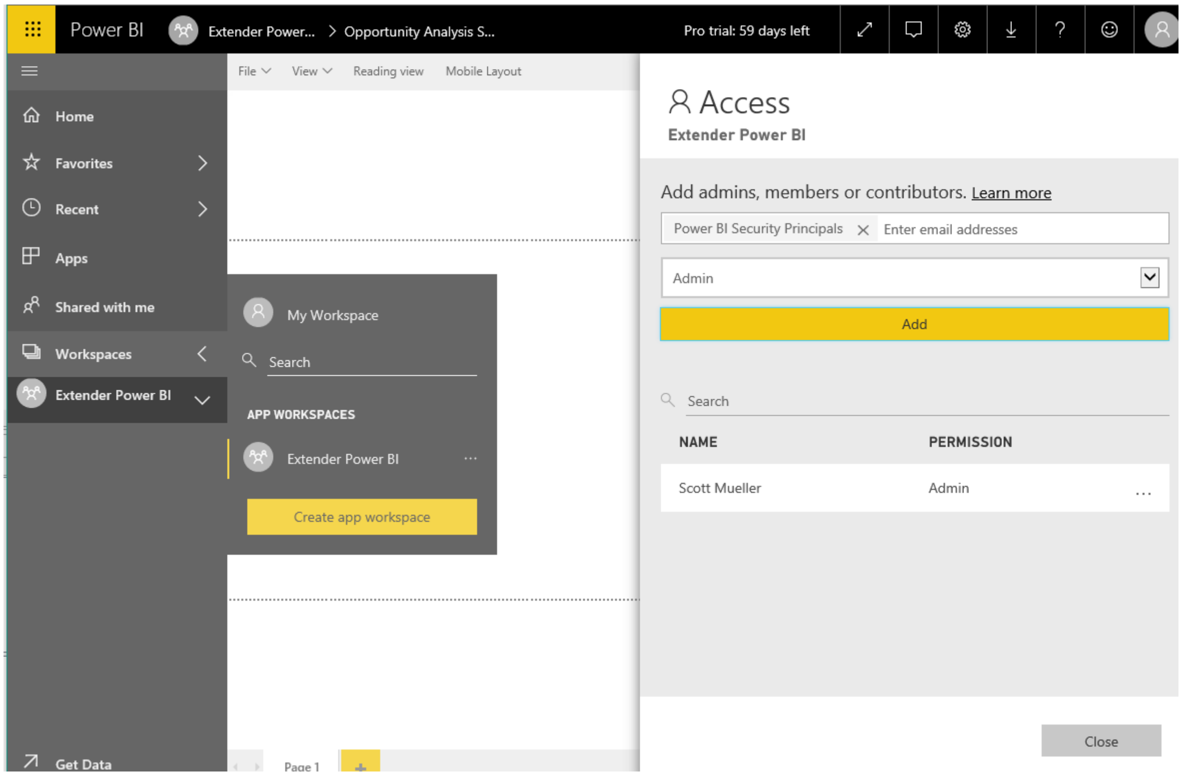
Po

Power BI Security Principals

Now select your Workspace and choose Workspace access (Note you must use a named Workspace and not "My Workspace"):

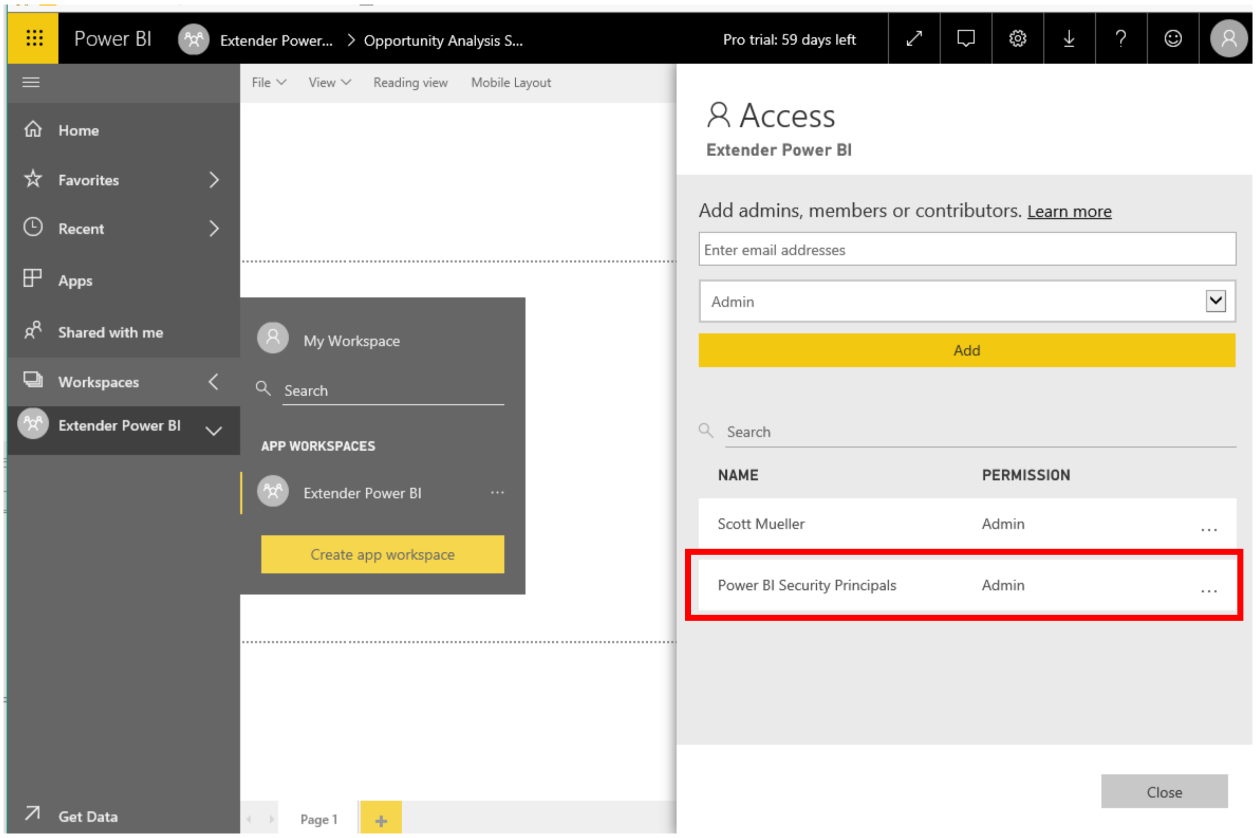


Add the name of the Security Group created above and add as an Admin:



NAME	PERMISSION
Scott Mueller	Admin

Verify the group appears in the list as Admin:

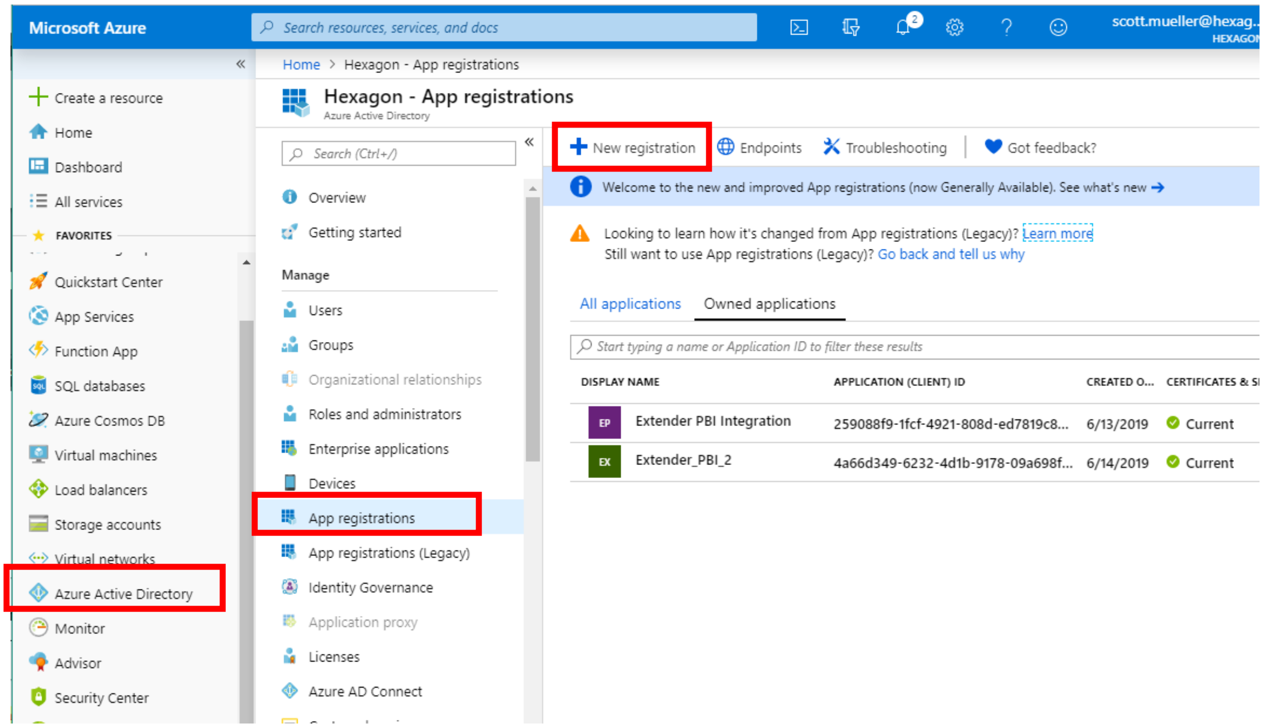


NAME	PERMISSION
Scott Mueller	Admin
Power BI Security Principals	Admin

Your Power BI application should now be ready to create a Power BI Data Source over it. When creating the Data Source, leave the Primary User Id / Primary Password empty.

User-Owns-Data

Go to Azure AD (https://portal.azure.com). Go to Azure Active Directory and click on App registrations, then select New Registration:

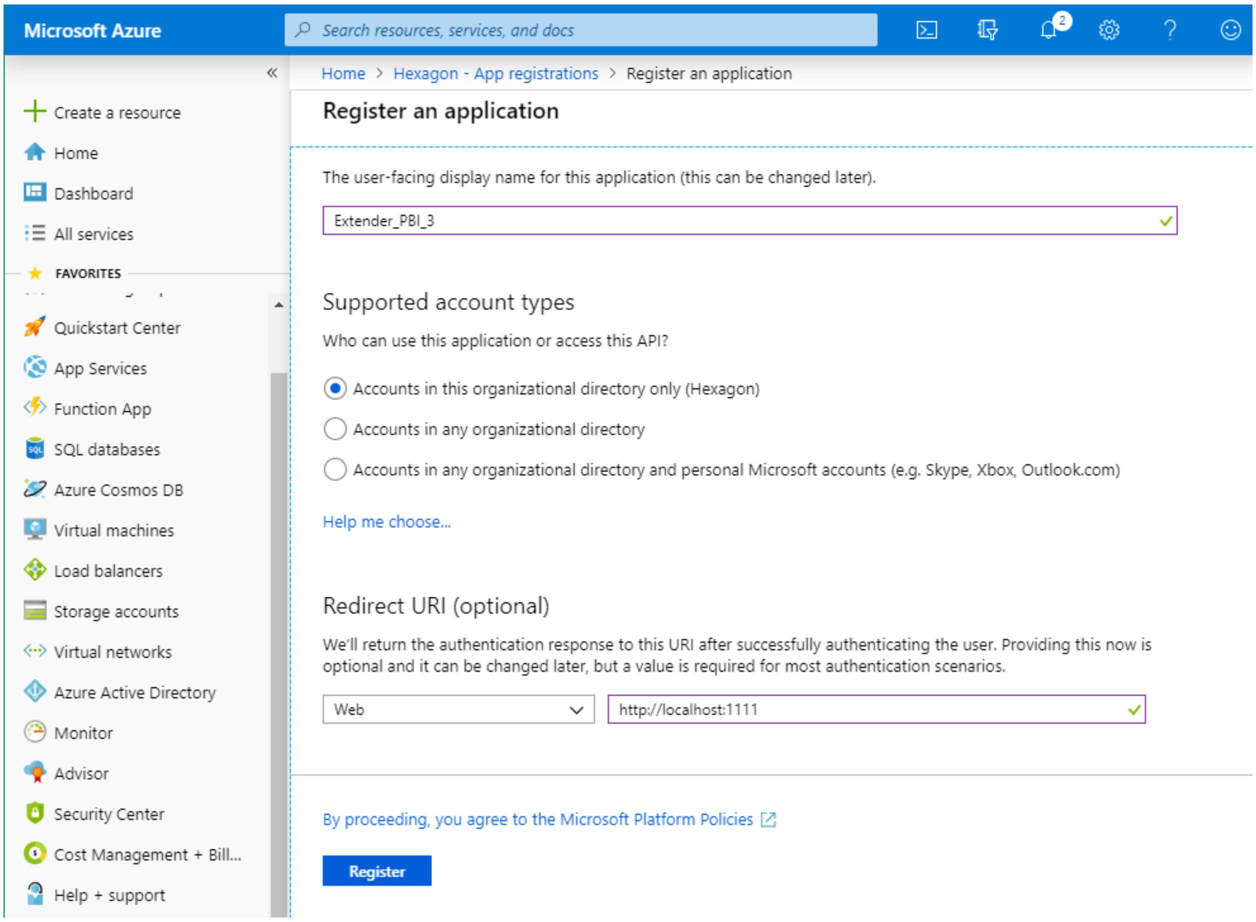


Welcome to the new and improved App registrations (now Generally Available). See what's new →

Looking to learn how it's changed from App registrations (Legacy)? Learn more
Still want to use App registrations (Legacy)? Go back and tell us why

DISPLAY NAME	APPLICATION (CLIENT) ID	CREATED O...	CERTIFICATES & SE
EP Extender PBI Integration	259088f9-1fcf-4921-808d-ed7819c8...	6/13/2019	✓ Current
EX Extender_PBI_2	4a66d349-6232-4d1b-9178-09a698f...	6/14/2019	✓ Current

Enter the application display name and Redirect URI. Note that you may enter any value for Redirect URI, as it will not be referenced by the Xalt application.



Register an application

The user-facing display name for this application (this can be changed later).

Extender_PBI_3 ✓

Supported account types

Who can use this application or access this API?

● Accounts in this organizational directory only (Hexagon)

○ Accounts in any organizational directory

○ Accounts in any organizational directory and personal Microsoft accounts (e.g. Skype, Xbox, Outlook.com)

Help me choose...

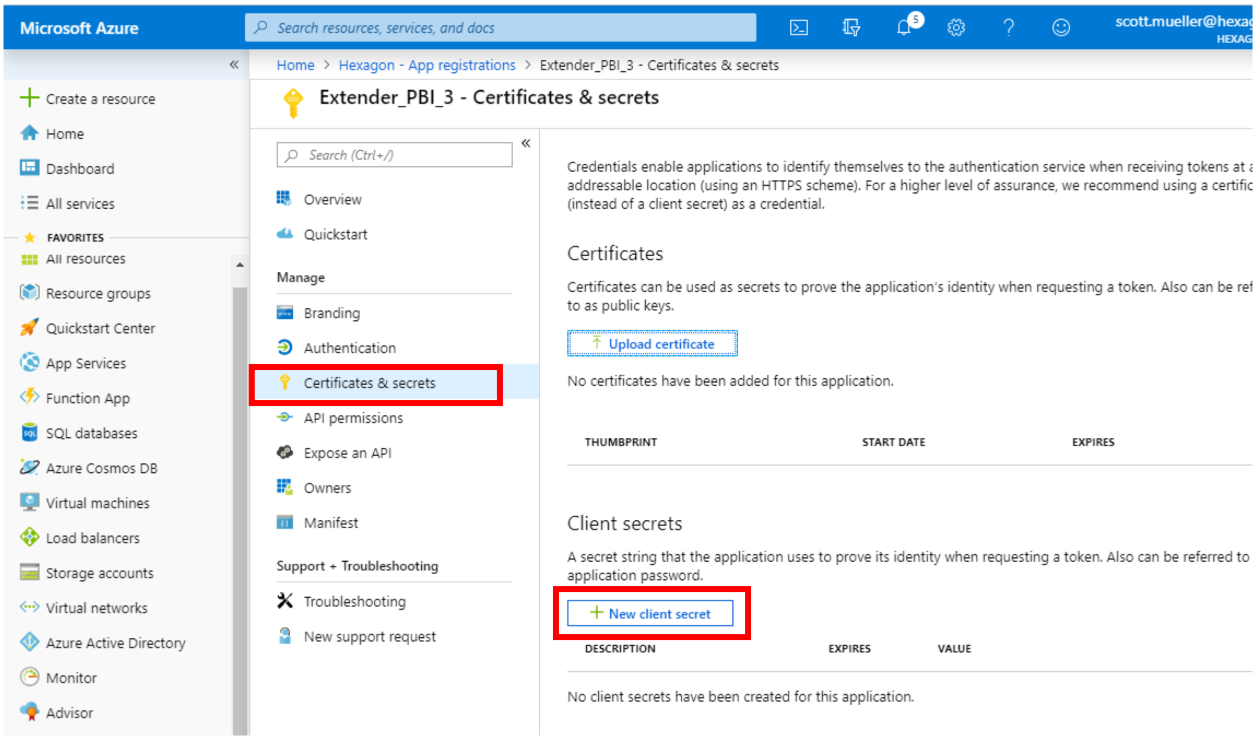
Redirect URI (optional)

We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Web | http://localhost:1111 ✓

By proceeding, you agree to the Microsoft Platform Policies

Next, go to Certificates and secrets and select New client secret:



Extender_PBI_3 - Certificates & secrets

Credentials enable applications to identify themselves to the authentication service when receiving tokens at a addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certific (instead of a client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be ref to as public keys.

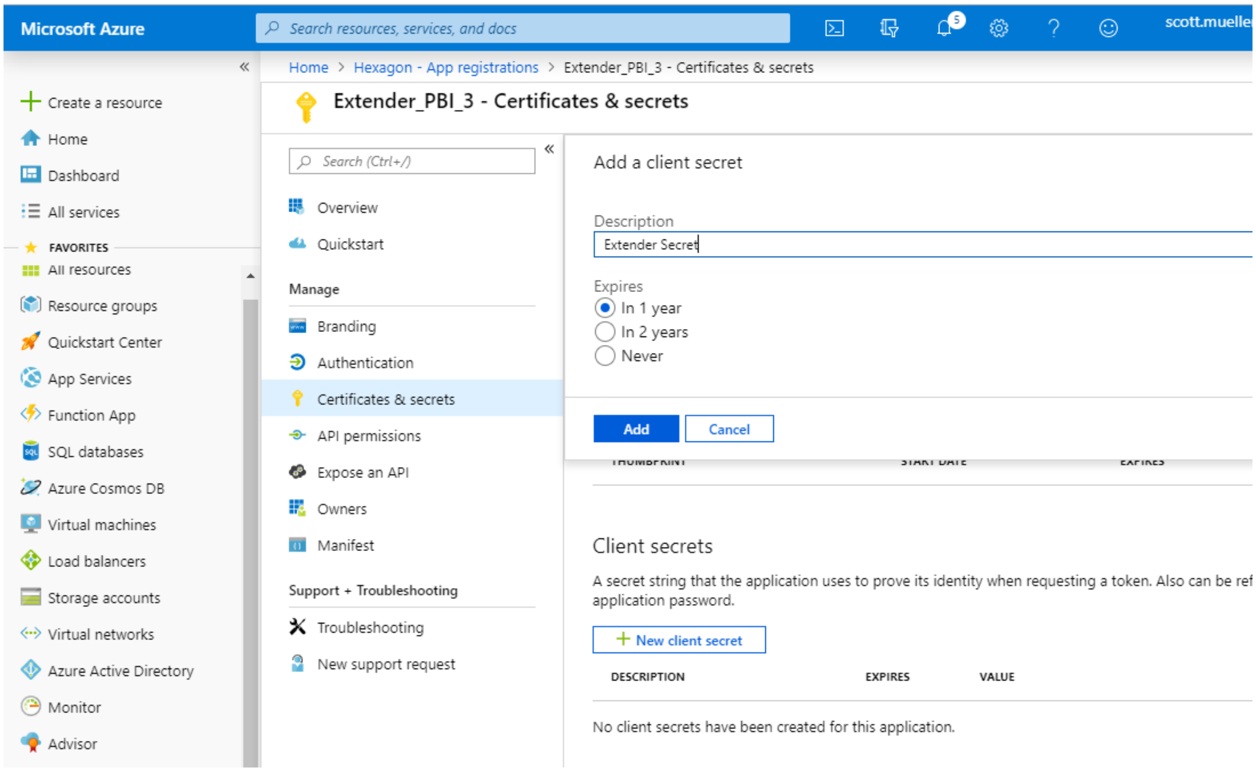
No certificates have been added for this application.

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred to application password.

No client secrets have been created for this application.

Enter the information to add a client secret. You may set an Expires value of your choosing:



Extender_PBI_3 - Certificates & secrets

Add a client secret

Description: Extender Secret

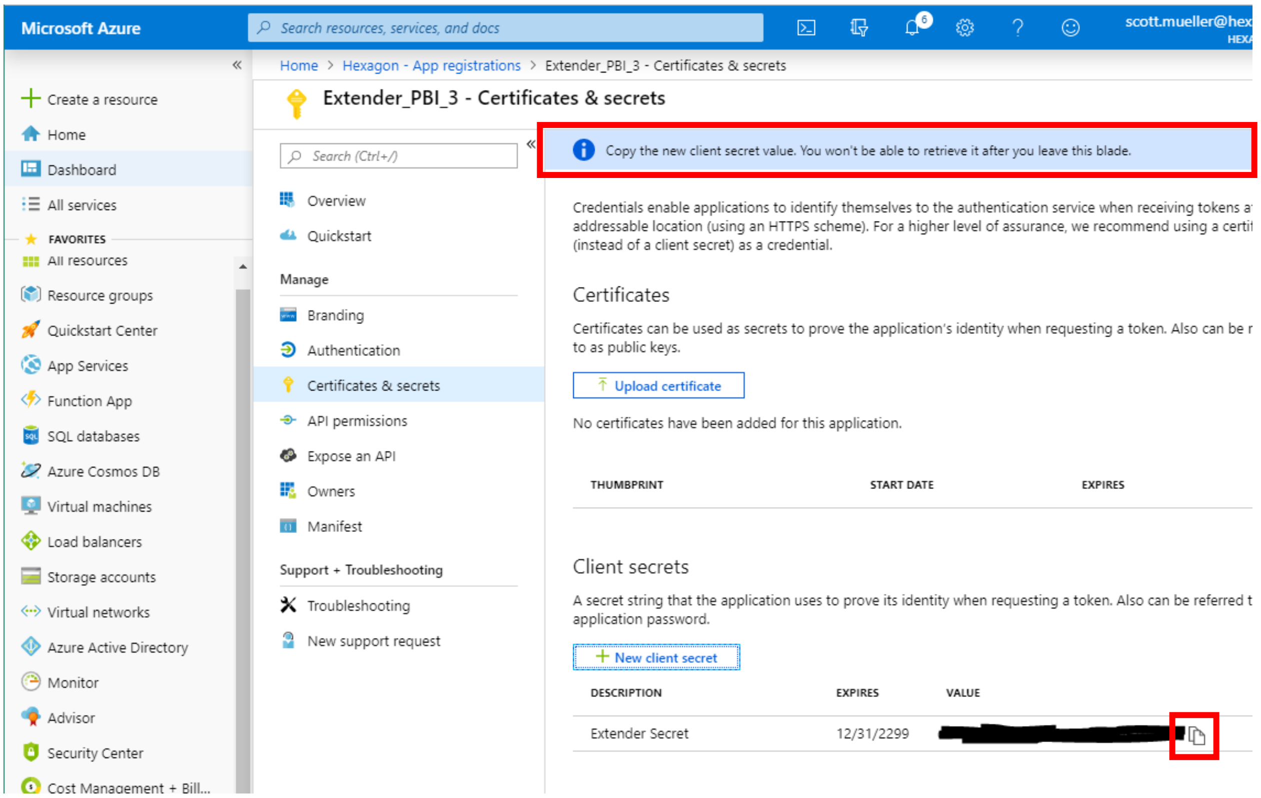
Expires
● In 1 year
○ In 2 years
○ Never

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be ref application password.

No client secrets have been created for this application.

Make sure to copy the value, as you cannot retrieve it again after it has been created:



Extender_PBI_3 - Certificates & secrets

Copy the new client secret value. You won't be able to retrieve it after you leave this blade.

Credentials enable applications to identify themselves to the authentication service when receiving tokens a addressable location (using an HTTPS scheme). For a higher level of assurance, we recommend using a certi (instead of a client secret) as a credential.

Certificates

Certificates can be used as secrets to prove the application's identity when requesting a token. Also can be r to as public keys.

No certificates have been added for this application.

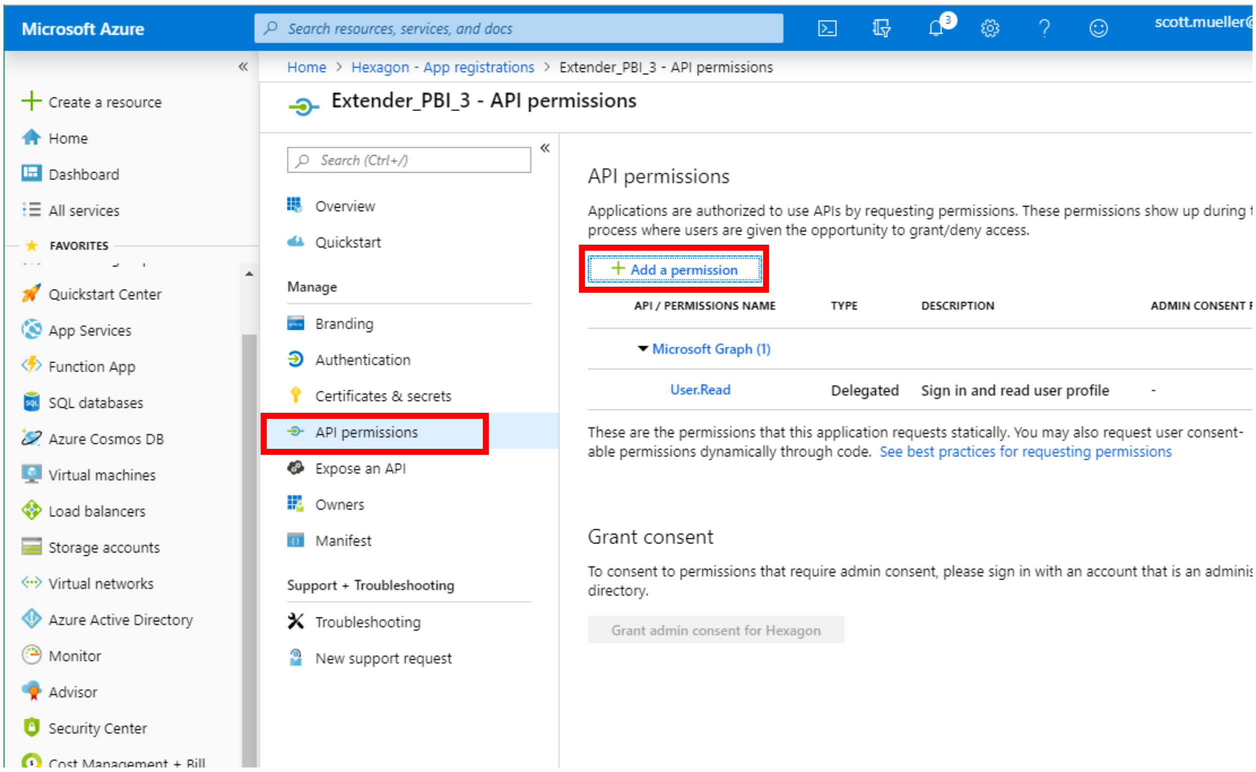
THUMBPRINT	START DATE	EXPIRES

Client secrets

A secret string that the application uses to prove its identity when requesting a token. Also can be referred t application password.

DESCRIPTION	EXPIRES	VALUE
Extender Secret	12/31/2299	[redacted]

Next, go to API Permissions and choose Add a permission:



Extender_PBI_3 - API permissions

API permissions

Applications are authorized to use APIs by requesting permissions. These permissions show up during t process where users are given the opportunity to grant/deny access.

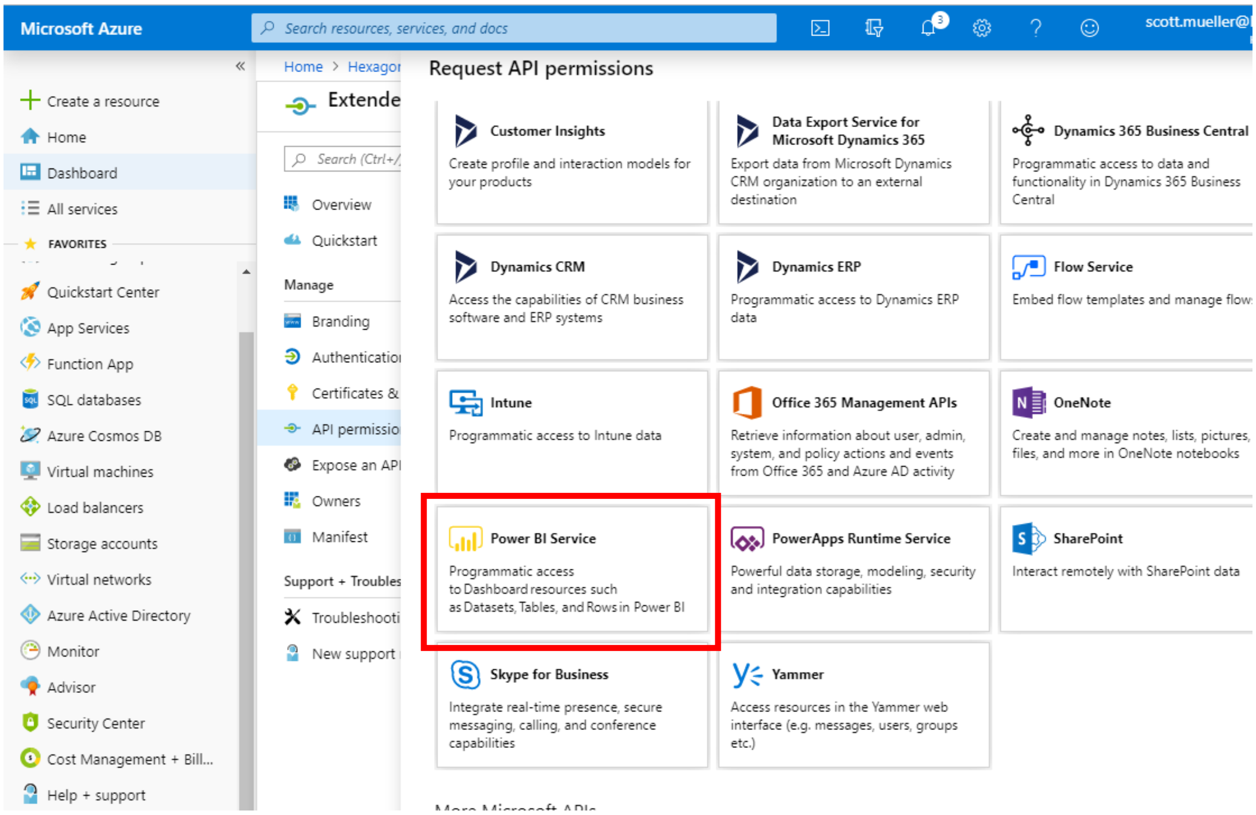
API / PERMISSIONS NAME	TYPE	DESCRIPTION	ADMIN CONSENT R
▾ Microsoft Graph (1)			
User.Read	Delegated	Sign in and read user profile	-

These are the permissions that this application requests statically. You may also request user consent-able permissions dynamically through code. See best practices for requesting permissions

Grant consent

To consent to permissions that require admin consent, please sign in with an account that is an adminis directory.

Select Power BI Service:



Request API permissions

Customer Insights – Create profile and interaction models for your products

Data Export Service for Microsoft Dynamics 365 – Export data from Microsoft Dynamics CRM organization to an external destination

Dynamics 365 Business Central – Programmatic access to data and functionality in Dynamics 365 Business Central

Dynamics CRM – Access the capabilities of CRM business software and ERP systems

Dynamics ERP – Programmatic access to Dynamics ERP data

Flow Service – Embed flow templates and manage flow

Intune – Programmatic access to Intune data

Office 365 Management APIs – Retrieve information about user, admin, system, and policy actions and events from Office 365 and Azure AD activity

OneNote – Create and manage notes, lists, pictures, files, and more in OneNote notebooks

Power BI Service – Programmatic access to Dashboard resources such as Datasets, Tables, and Rows in Power BI

PowerApps Runtime Service – Powerful data storage, modeling, security and integration capabilities

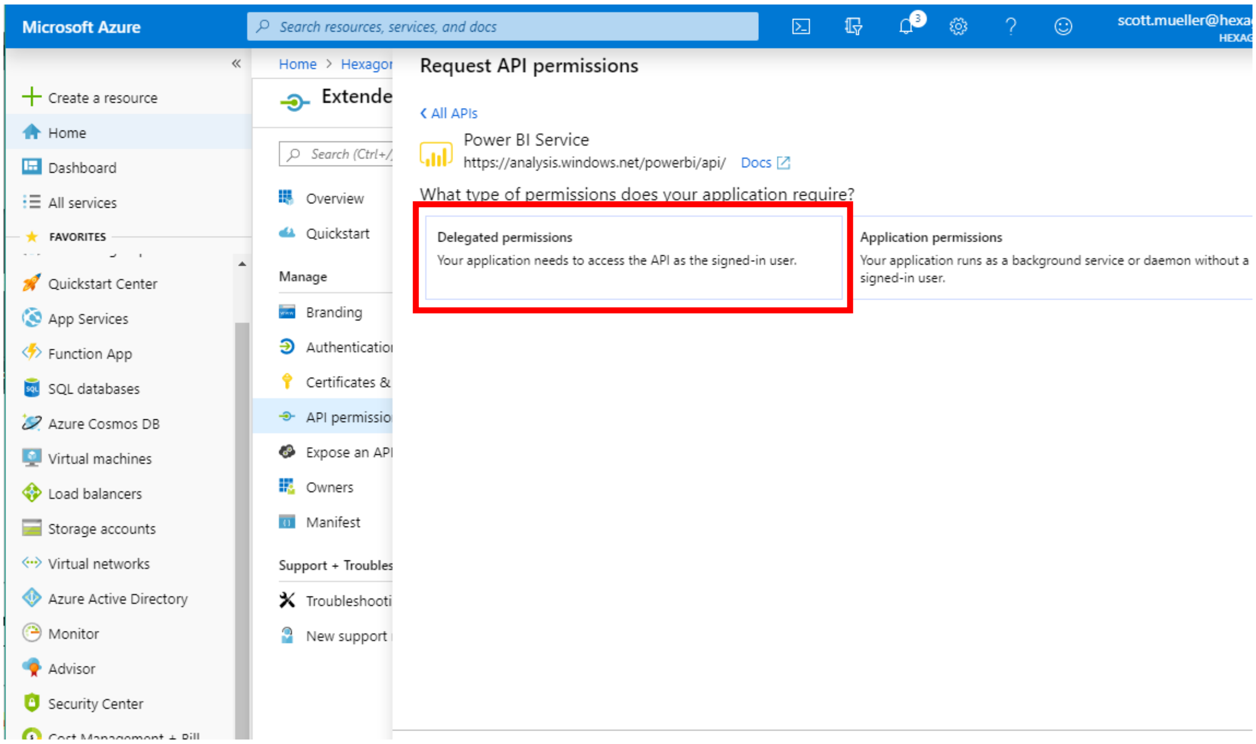
SharePoint – Interact remotely with SharePoint data

Skype for Business – Integrate real-time presence, secure messaging, calling, and conference capabilities

Yammer – Access resources in the Yammer web interface (e.g. messages, users, groups etc.)

More Microsoft APIs

Select Delegated permissions:



Request API permissions

< All APIs

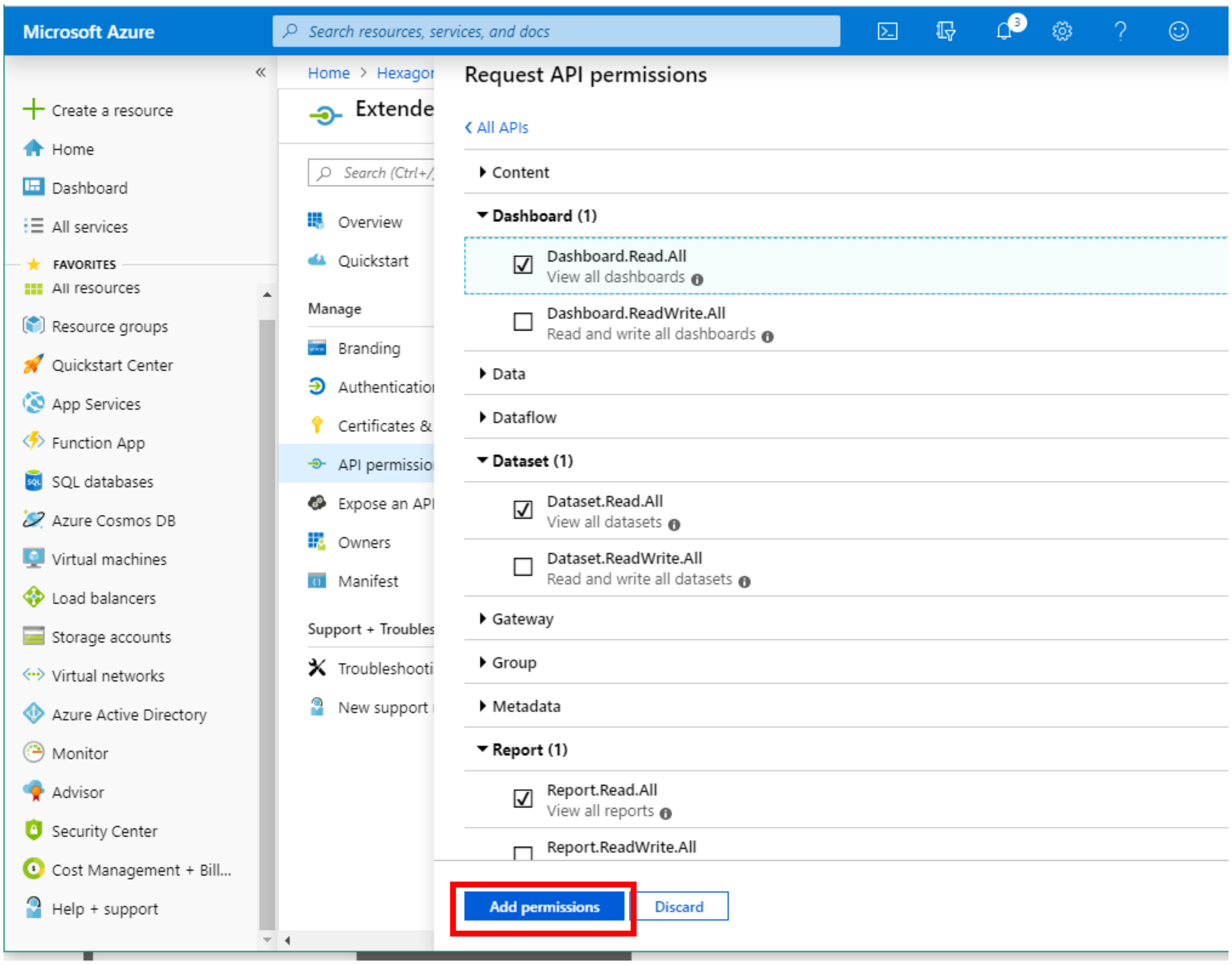
Power BI Service
https://analysis.windows.net/powerbi/api/ Docs

What type of permissions does your application require?

Delegated permissions – Your application needs to access the API as the signed-in user.

Application permissions – Your application runs as a background service or daemon without a signed-in user.

Select Dashboard.Read.All, Dataset.Read.All, and Report.Read.All permissions (note that Dashboard.Read.All is not currently required but may be useful in the future if Xalt is enhanced to support Dashboards as well as Reports):

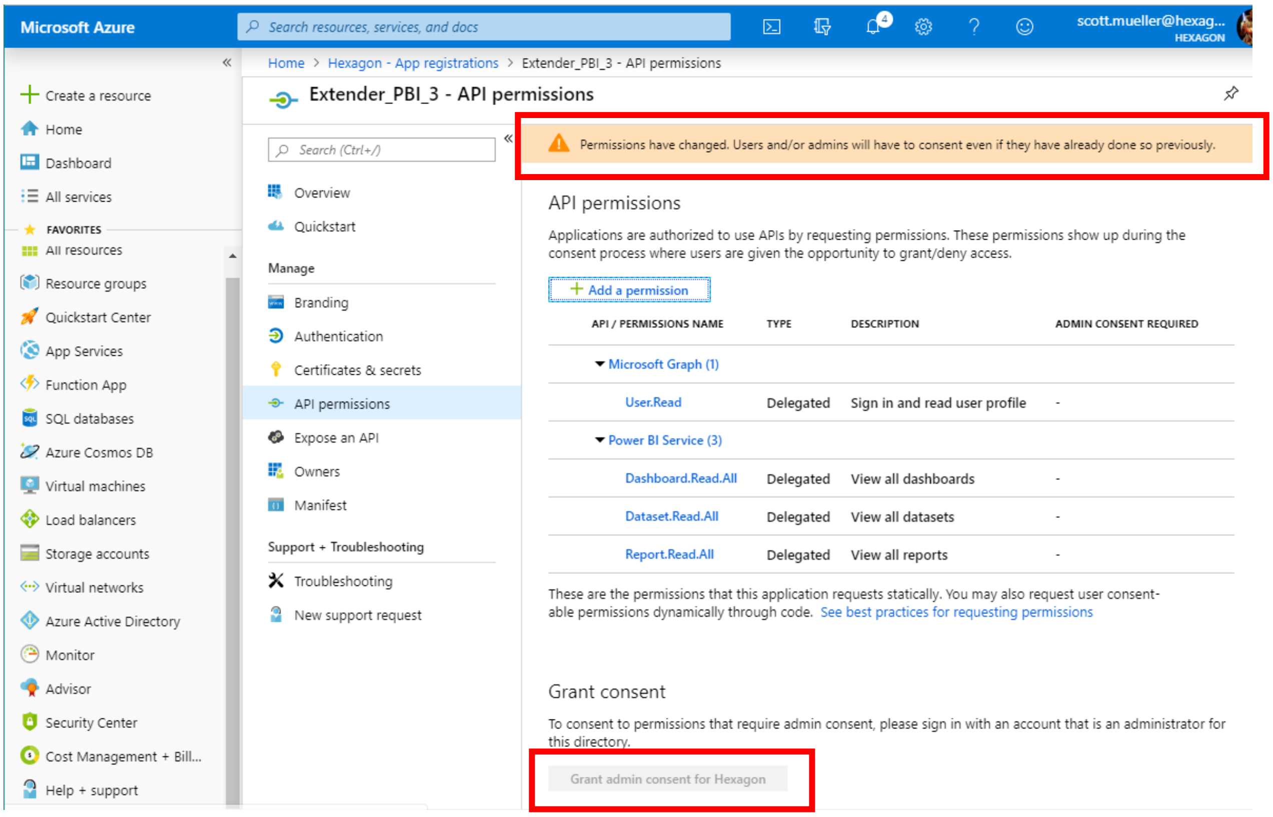


Request API permissions

‹ All APIs

- ▸ Content
- ▾ Dashboard (1)
  - ☑ Dashboard.Read.All — View all dashboards
  - ☐ Dashboard.ReadWrite.All — Read and write all dashboards
- ▸ Data
- ▸ Dataflow
- ▾ Dataset (1)
  - ☑ Dataset.Read.All — View all datasets
  - ☐ Dataset.ReadWrite.All — Read and write all datasets
- ▸ Gateway
- ▸ Group
- ▸ Metadata
- ▾ Report (1)
  - ☑ Report.Read.All — View all reports
  - ☐ Report.ReadWrite.All

Note that you get a message saying consent is required, but the Consent button may be disabled:



Extender_PBI_3 - API permissions

Permissions have changed. Users and/or admins will have to consent even if they have already done so previously.

API permissions

Applications are authorized to use APIs by requesting permissions. These permissions show up during the consent process where users are given the opportunity to grant/deny access.

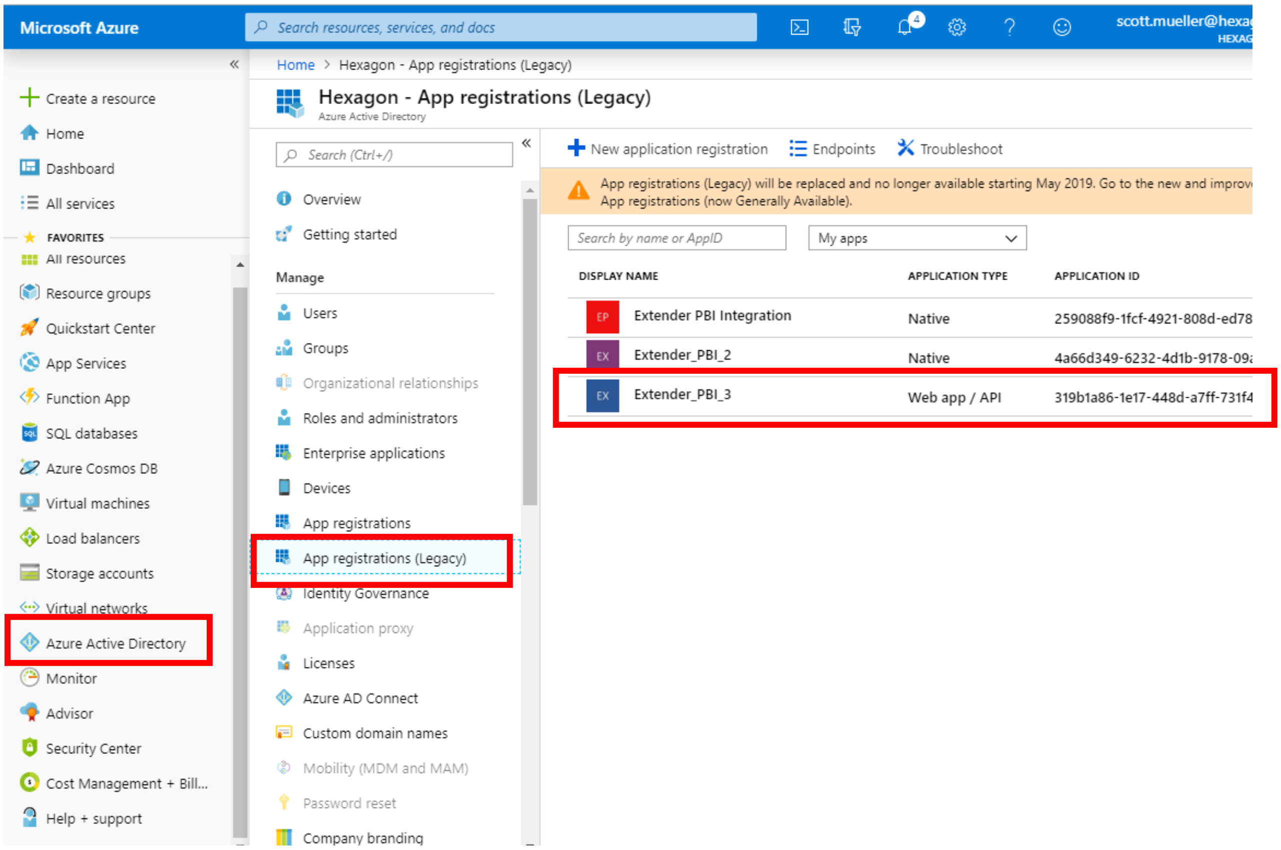
API / PERMISSIONS NAME	TYPE	DESCRIPTION	ADMIN CONSENT REQUIRED
Microsoft Graph (1)			
User.Read	Delegated	Sign in and read user profile	-
Power BI Service (3)			
Dashboard.Read.All	Delegated	View all dashboards	-
Dataset.Read.All	Delegated	View all datasets	-
Report.Read.All	Delegated	View all reports	-

These are the permissions that this application requests statically. You may also request user consent-able permissions dynamically through code. See best practices for requesting permissions

Grant consent

To consent to permissions that require admin consent, please sign in with an account that is an administrator for this directory.

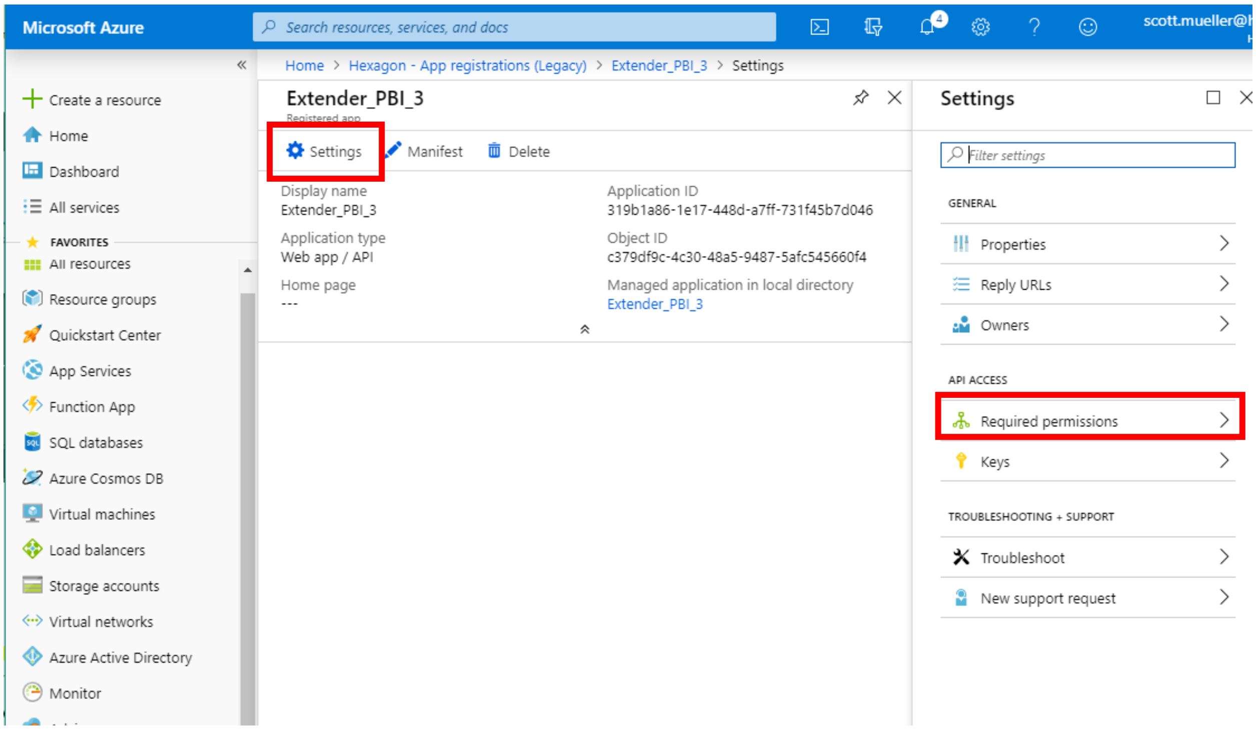
In this case, go to Azure Active Directory, App registration (Legacy), and select your application:



App registrations (Legacy) will be replaced and no longer available starting May 2019. Go to the new and improv
App registrations (now Generally Available).

DISPLAY NAME	APPLICATION TYPE	APPLICATION ID
EP Extender PBI Integration	Native	259088f9-1fcf-4921-808d-ed78
EX Extender_PBI_2	Native	4a66d349-6232-4d1b-9178-09a
EX Extender_PBI_3	Web app / API	319b1a86-1e17-448d-a7ff-731f4

Now select Settings, Required permissions:



Display name
Extender_PBI_3

Application type
Web app / API

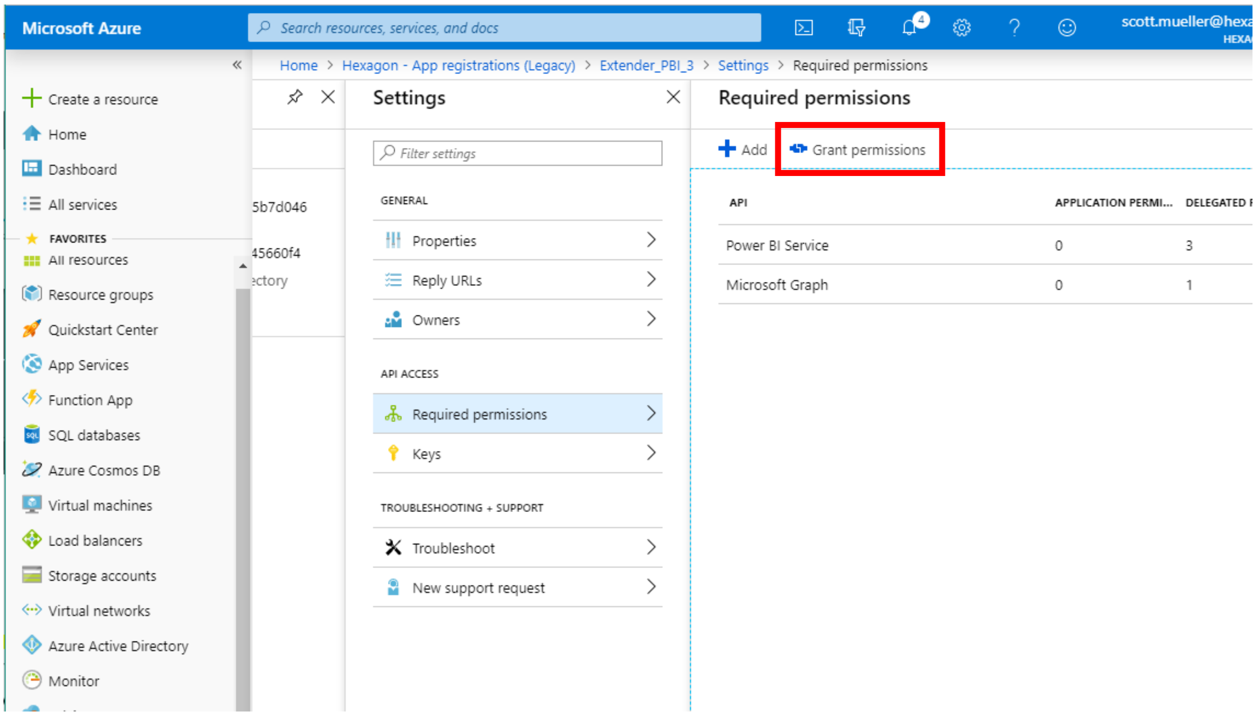
Home page

Application ID
319b1a86-1e17-448d-a7ff-731f45b7d046

Object ID
c379df9c-4c30-48a5-9487-5afc545660f4

Managed application in local directory
Extender_PBI_3

Click the Grant permissions button:



API	APPLICATION PERMI...	DELEGATED P
Power BI Service	0	3
Microsoft Graph	0	1

Your Power BI application should now be ready to create a Power BI Data Source over it. When creating the Data Source, set the Primary User Id / Primary Password to the Power BI User that owns the Workspace you want to embed reports from.

Document Change History

April 13, 2023

- Added information for Current Failed Logins section of Administrator Dashboard
    - Chapter 9 – User Profile Administration
- Added information for new multi-Request feature
    - Chapter 6 – Data Object Actions

September 16, 2022

- Added information for allowing draggable Gantt Chart bars
    - Chapter 4 – Data Object Queries

July 29, 2022

- Added information for new Calendar Week Start Custom Setting
    - Chapter 12 – Custom Settings
- Added information for new Password Requirements Custom Settings
    - Chapter 12 – Custom Settings
- Added information for Password Expiration Interval
    - Chapter 9 – User Profile Administration

June 29, 2022

- Added information for new GML for XHA Custom Setting
    - Chapter 12 – Custom Settings

February 4, 2022

- Added information for new Date Presentation Format function
    - Chapter 3 – Data Objects
    - Chapter 12 – Custom Settings

August 16, 2021

- Added information for new NFC Scanning function
  - Chapter 3 – Data Objects
  - Chapter 6 – Data Object Actions
  - Appendix A – Specifying Messages and Substitution Values

June 25, 2021

- Added information for new submit Import Objects to batch feature
  - Chapter 15 – Import/Export

May 24, 2021

- Added information for new Gantt Chart Query feature
  - Chapter 4 – Data Object Queries
- Added information for new ODATA Detail Visualization Section
  - Chapter 5 – Data Object Details
- Added information for new Hexagon PPM Visualization Section feature
  - Chapter 6 – Data Object Actions
- Added information for new Local Session Properties
  - Appendix A – Specifying Messages and Substitution Values
- Added information for new Maintainable Query feature
  - Chapter 4 – Data Object Queries
  - Chapter 14 – Dashboards

January 20, 2021

- Added information for new Package Log Files option for Connector gateways
  - Chapter 2 – Connector gateways and Data Sources
- Added information for new Allow Quick Search value on Data Objects
  - Chapter 3 – Data Objects

September 3, 2020

- Added information for new feature to specify single/double click to open list records in XHA
  - Chapter 12 – Custom Settings

June 30, 2020

- Added information for new feature to Hide Find Toolbar action on Object Lookups
  - Chapter 3 – Data Objects

June 29, 2020

- Added information for new feature to Hide custom/maintenance Actions on specific Query/Details
  - Chapter 4 – Data Object Queries
  - Chapter 5 – Data Object Details

April 22, 2020

- Added information for new Hexagon PPM EDE (Engineering Data Editor) Views feature
  - Chapter 6 – Data Object Actions

April 13, 2020

- Added information for new Reassign Domain Class feature
  - Chapter 3 – Data Objects
  - Chapter 15 – Import/Export

April 3, 2020

- Added information for new Donut Chart feature
  - Chapter 4 – Data Object Queries

February 15, 2020

- Added information for new feature to skip processing of ExecuteHttpAction response
  - Chapter 6 – Data Object Actions

January 22, 2020

- Added information for new feature to Hide Actions on specific Query/Details
  - Chapter 4 – Data Object Queries
  - Chapter 5 – Data Object Details

December 19, 2019

- Added information for new Generic Offline feature
  - Chapter 3 – Data Objects
  - Chapter 6 – Data Object Actions
  - Chapter 14 – Dashboards

November 5, 2019

- Added information for new Allow use of Parent Paths for SELECTED_ODATA_OBJECT_ID
  - Appendix E - Mapping ODATA $metadata to Xalt

June 25, 2019

- Added information for new Microsoft Power BI Integration
  - Chapter 1 – Concepts
  - Chapter 2 – Connector gateways and Data Sources
  - Chapter 3 – Data Objects
  - Chapter 7 – Workbenches
  - Appendix F - Microsoft Power BI Integration
- Added information for new URL Encoding function
  - Chapter 6 – Data Object Actions

June 7, 2019

- Added information for new Clearing of Selected Value function
  - Chapter 3 – Data Objects
  - Chapter 6 – Data Object Actions

April 12, 2019

- Added information for new Auto Refresh function
  - Chapter 3 – Data Objects
  - Chapter 8 – Gadgets
  - Chapter 14 – Dashboards

March 8, 2019

- Added information for new Data Object Import Issues
  - Chapter 15 – Import/Export
- Added information for Synchronize Workbenches function
  - Chapter 7 – Workbenches

February 27, 2019

- Added Data Source Maintenance Permissions
  - Chapter 2 – Connector gateways and Data Sources
  - Chapter 10 – Security

December 4, 2018

- Rebranding of Hexagon to Xalt
  - Chapter 9 – User Profile Administration
  - Chapter 16 -- Tenant Email Server

November 7, 2018

- Added Premium Users to Organizations
    - Chapter 9 – User Profile Administration
    - Appendix A -- Appendix A -- Specifying Messages and Substitution Values

October 30, 2018

- Rebranded Hexagon Cloud Database Data Source type to Xalt Cloud Database
    - Chapter 2 – Connector gateways and Data Sources

September 17, 2018

- Added information for ODATA Logical Properties and Completion Actions
    - Appendix E – Mapping ODATA $metadata to Xalt

August 10, 2018

- Added information for support of Multi-Select Actions
    - Chapter 6 – Data Object Actions
- Added information for LATITUDE_RAW and LONGITUDE_RAW values for CURRENT_LOCATION and SELECTED_LOCATION Substitution Values
    - Appendix A – Specifying Messages and Substitution Values

July 10, 2018

- Added information for using Summary Queries for Object Lookups
    - Appendix C – Summary (Group-By) Queries

May 15, 2018

- Added information for ${CURRENT_USER_ORGANIZATION_DESCRIPTION} and ${CURRENT_USER_ORGANIZATION_ROLE} Substitution Values
    - Appendix A – Specifying Messages and Substitution Values
- Extended Client Inactivity Timeout maximum to 240 minutes
    - Chapter 12 – Custom Settings
- Added Information for allowing alternate FTP ports
    - Chapter 2 – Connector gateways and Data Sources

January 28, 2018

- Added information for new ODATA Data Source type
    - Chapter 2 – Connector gateways and Data Sources
    - Appendix E – Mapping ODATA $metadata to Xalt
- Added information for Request XML enhancements for ODATA

    - Chapter 6 – Data Object Actions
- Added information for maintaining Default Properties in an Action
    - Chapter 6 – Data Object Actions

January 22, 2018

- Added information for new Action Substitution Values and Data Source Substitution Values
    - Appendix A – Specifying Messages and Substitution Values
- Added information for Defined section to Default Properties
    - Appendix A – Specifying Messages and Substitution Values
- Added information for exposing Hexagon Attachments to the REST API
    - Appendix D – Attachments

November 20, 2017

- Added information for Organizations
    - Chapter 9 – User Profile Administration

August 22, 2017

- Added information for new Hide Unauthorized Actions value on Data Objects
    - Chapter 3 – Data Objects
- Added information for new **Hidden** value on Data Object Queries and Details
    - Chapter 4 – Data Object Queries
    - Chapter 5 – Data Object Details
- Added information for new **Hide Other Queries/Details** value
    - Chapter 3 – Data Objects
    - Chapter 4 – Data Object Queries
    - Chapter 5 – Data Object Details
    - Chapter 6 – Data Object Actions (and here)
    - Chapter 11 – Business Intelligence (BI)
    - Chapter 14 – Dashboards
- Added information for new Copy function on Security Roles
    - Chapter 10 – Security

August 1, 2017

- Added information for new Override Label value on Actions
    - Chapter 6 – Actions

June 20, 2017

- Added information for Resolve Common Issues action (replaces Reassign Reference action)
    - Chapter 15 – Import/Export

May 16, 2017

- Added information for Authenticating Users List
    - Chapter 2 – Connector gateways & Data Sources
- Added information for new Substitution Functions
    - Appendix A – Specifying Messages and Substitution Values

May 3, 2017

- Added information for Tenant Email Server
    - Chapter 16 – Tenant Email Server

April 4, 2017

- Added information for assigned Security Role Workbenches to User Profiles:
    - Chapter 7 – Workbenches
    - Chapter 9 – User Profile Administration
    - Chapter 10 – Security

- Added information for Reassign Reference action and new Data Source issues:
    - Chapter 15 – Import/Export

March 20, 2017

- Chapter 12 – Custom Settings
    - Added information for Suppress Diagnostic Info

February 20, 2017 – changes since August 22, 2016

- Chapter 2 – Connector gateway and Data Sources
    - Removed Enter GML Development Mode section on page 42
- Chapter 3 – App Builder Dashboard
    - Figure 4 updated
    - Updated Enter Development Mode section
    - Added Exit Development Mode section
- Chapter 6 – Data Object Actions
    - Added Override Standard Action section